Goals

- Laid out a rough outline of how SHAP would be computed, I thought I would give SHAP methods a try
- Earlier methods work and prove that the model is unpickled and can be used
- Be able to iterate through each trained model to it's respective CV dataset, create shap values, generate shap plots
- Be able to store each CV shap values for each model and store in csv file as a DataFrame

```
            LR_shap_all_CVs.csv ==>
                                      LR_0 --> CV0
                                      LR_1 --> CV1
                                      LR_2 --> CV2
```

Things to do:

- Still need to figure out saving results into a file (pickle.dump()), create and save into designated folder
- Figure out how to work TreeExplainer, expected_value function
- Find file with the feature names for corresponding dataset to load into program under 'Load Metadata" section
- Figure out how to display other shap plots such as waterfall, force plot, etc

Notes

- Most of the program is hardcoded to specifically load one of the trained models after running STREAMLINE **resolved**
- Was able to prove that the model can be unpickled and used for .predict() and .predict*proba*() **resolved**
- Was able to use model to create SHAP explainers, calculate shap_values for CV0 testing dataset, and display plots **resolved**
- However, still need to refine the SHAP methods as there were some issues for Decision Tree Classifier **resolved**
- Was able to display Decision Tree prediction using TreeExplainer or even Explainer....I might be doing something wrong **resolved**
- XGBOOST MODEL IS COMPATIBLE WITH ALL OF THE LISTED SHAP PLOTS **resolved**
- RF MODEL NEEDED IT'S OWN IF-STATEMENT FOR NOW BUT WILL CONDENSE FOR CLARITY ADN EFFICIENCY **resolved**
- STILL NEED TO WORK ON LIGHTGBM, CATBOOST **resolved**
- GO BACK TO FIX DECISION TREE **resolved**

Fix

- Go back to double check shap plot compatibility for global and local importance for linear models **resolved**
- Work through the DecisionTreeClassifier and compare to other codes out there (if possible) **resolved**
- Currently unsure if creating dataframe for each model's shap_values shuold be done in compute_shap_values() or within the nested for-loop in testing cell
- Feature names when displaying shap plots

Updates (refer to 'Next Steps' for more updates)

7/29/22

- ALL given SHAP plots seems to work for NB() when not in a defined function block and if-statement **resolved**
- Bar, scatter, waterfall, and beeswarm plots don't work for LR(), other plots work fine on LinearExplainer() and shap_values = explainer.shap_values(data)

8/02/22

- Plots and shap_values for each trained model in each CV work
- Will focus on section called **'Next Steps'**
    - refer to bottom of Notebook for more details
    - Currently unsure if creating dataframe for each model's shap_values shuold be done in compute_shap_values() or within the nested for-loop in testing cell

In [1]:

```
# required packages & models
import os
import sys
import pickle
import warnings
warnings.filterwarnings('ignore')
import csv
import sklearn
import shap
import numpy as np
import numpy.typing as npt
import pandas as pd
import scipy as sp
import matplotlib.pyplot as plt
from matplotlib.backends.backend_pdf import PdfPages
from fpdf import FPDF
import collections
from termcolor import colored as cl #text customization

# Model packages
import xgboost
import lightgbm as lgb
from sklearn import *
from sklearn.naive_bayes import GaussianNB
from sklearn.linear_model import LogisticRegression
from sklearn.tree._classes import DecisionTreeClassifier
from sklearn.ensemble import RandomForestClassifier
import xgboost as xgb
import lightgbm as lgb
import catboost as cgb
from sklearn import tree
from shap.plots import waterfall

#import metrics
from sklearn.model_selection import train_test_split
from sklearn.metrics import classification_report, accuracy_score

# Jupyter Notebook Hack: This code ensures that the results of multiple commands within a given cell are all displayed
from IPython.core.interactiveshell import InteractiveShell
InteractiveShell.ast_node_interactivity = "all"


shap.initjs() # load JS visualization code to notebook. SHAP plots won't be displayed without this
```

Run Parameters

In [2]:
```
dataset_path = "/Users/jessicakim/Desktop/STREAMLINE/DemoData"
experiment_path = "/Users/jessicakim/Desktop/STREAMLINE/DemoData/Output/hcc_demo"
targetDataName = 'None'

# hardcoded pathways for CVDataset0
# train_file_path = '/hcc-data_example/CVDatasets/'
# test_file_path = '/hcc-data_example/CVDatasets/'
```

Check for Analyzed Datasets and Remove Unecessary Files

In [3]:
```
datasets = os.listdir(experiment_path)
experiment_name = experiment_path.split('/')[-1] #Name of experiment folder

datasets.remove('metadata.csv')
datasets.remove('metadata.pickle')
datasets.remove('algInfo.pickle')

try:
    datasets.remove('jobsCompleted')
except:
    pass
try:
    datasets.remove('UsefulNotebooks')
except:
    pass
try:
    datasets.remove('logs')
    datasets.remove('jobs')
except:
    pass
try:
    datasets.remove('DatasetComparisons') #If it has been run previously (overwrite)
except:
    pass
try:
    datasets.remove('KeyFileCopy') #If it has been run previously (overwrite)
except:
    pass
try:
    datasets.remove('.DS_Store') #If it has been run previously (overwrite)
except:
    pass
try:
    datasets.remove(experiment_name+'_ML_Pipeline_Report.pdf') #If it has been run previously (overwrite)
except:
    pass

datasets = sorted(datasets) #ensures consistent ordering of datasets
print("Analyzed Datasets: "+str(datasets))
```

```
Analyzed Datasets: ['hcc-data_example', 'hcc-data_example_no_covariates']
```

Load Metadata and Other Necessary Variables

In [4]:

```
jupyterRun = 'True'
# Loading necessary variables specified earlier in the pipeline from metadatafor dataPrep()
file = open(experiment_path + '/' + "metadata.pickle", 'rb')
metadata = pickle.load(file)
# file.close()
# print(metadata)

class_label = metadata['Class Label']
instance_label = metadata['Instance Label']
cv_partitions = int(metadata['CV Partitions'])



# # # unpickle and load in feature_names found in 'categorical_variables.pickle'
# feature_names_file = experiment_path + '/hcc-data_example/exploratory/categorical_variables.pickle'
# file = open(feature_names_file , 'rb')
# feature_names= pickle.load(file)
# file.close()
# print('Checking for feature names...\n',feature_names)



alg_file = open(experiment_path + '/' + "/algInfo.pickle", 'rb')
algInfo = pickle.load(alg_file)
alg_file.close()
algorithms = []


abbrev = {}
for key in algInfo:  # pickling specific model while also checking for corresponding algInfo
    if algInfo[key][0]: # If that algorithm was used
        algorithms.append(key)
        abbrev[key] = (algInfo[key][1])

print('\nChecking for algorithms used in STREAMLINE...\n',algorithms)
print('\nChecking for abbrev for algorithms used in STREAMLINE...\n', abbrev)
```

```
Checking for algorithms used in STREAMLINE...
 ['Naive Bayes', 'Logistic Regression', 'Decision Tree', 'Random Forest', 'Extreme Gradient Boosting']

Checking for abbrev for algorithms used in STREAMLINE...
 {'Naive Bayes': 'NB', 'Logistic Regression': 'LR', 'Decision Tree': 'DT', 'Random Forest': 'RF', 'Extreme Gradient Bo
osting': 'XGB'}
```

Get Feature Names From Target Dataset

In [5]:

```
# user can choose which csv dataset file to use if more than one was analyzed
target_dataset = '/hcc-data_example.csv'    # default is 'None'
orig_dataset = dataset_path + '/' + target_dataset
# print(orig_dataset)

# feature_names = pd.read_csv(orig_dataset)
# if instance_label != 'None':
#     feature_names = feature_names.drop(instance_label,axis=1)
#     feature_names = feature_names.drop(class_label, axis= 1).columns
# print(feature_names)
```

dataPrep(): Loading Target CV Training & Testing Sets

In [7]:

```
def dataPrep(train_file_path,instance_label,class_label, test_file_path):

    '''Loads target cv training dataset, separates class from features and removes instance labels'''

    # store all used features from training and testing sets which will be later used for shapley valeues & plots
    train_feat = {}
    test_feat = {}

    train = pd.read_csv(train_file_path)
    if instance_label != 'None':
        train = train.drop(instance_label,axis=1)

    # populate train_feat[] with feature names from given train dataset
    train_feat = train.drop(class_label, axis = 1).columns.tolist()
    print(train_feat)

    trainX = pd.DataFrame(train.drop(class_label,axis=1).values)
    trainY = pd.DataFrame(train[class_label].values)
    del train #memory cleanup


    test = pd.read_csv(test_file_path)
    if instance_label != 'None':
        test = test.drop(instance_label,axis=1)

     # populate train_feat[] with feature names from given test dataset
    test_feat = test.drop(class_label, axis = 1).columns.tolist()
    print(test_feat)

    testX = pd.DataFrame(test.drop(class_label,axis=1).values)
    testY = pd.DataFrame(test[class_label].values)
    del test #memory cleanup



    return trainX, trainY, testX, testY, train_feat, test_feat
```

SHAP: get_explainer()

- will check if explainer is one of the available ML in STREAMLINE
- if algorithm name matches ['list model names'], create explainers
- return explainer based on given model from parameter

Types of SHAP Explainers

.Explainer()

- Uses Shapley values to explain any machine learning model or python function.
- This is the primary explainer interface for the SHAP library
- It takes any combinationof a model and masker and returns a callable subclass object that implements the particular estimation algorithm that was chosen.

.TreeExplainer()

- Uses Tree SHAP algorithms to explain the output of ensemble tree models.
- Tree SHAP is a fast and exact method to estimate SHAP values for tree models and ensembles of trees, under several different possible assumptions about feature dependence.
- It depends on fast C++implementations either inside an externel model package or in the local compiled C extention.

.LinearExplainer()

- Computes SHAP values for a linear model, optionally accounting for inter-feature correlations.
- This computes the SHAP values for a linear model and can account for the correlations among the input features.
- Assuming features are independent leads to interventional SHAP values which for a linear model are coef[i] * (x[i] - X.mean(0)[i]) for the ith feature.
- If instead we accountfor correlations then we prevent any problems arising from colinearity and share credit among correlated features.
- Accounting for correlations can be computationally challenging, but LinearExplainer uses sampling to estimate a transform that can then be applied to explain any prediction of the model.

In [9]:

```
def get_explainer(model, abbrev, trainX):

    '''Pass loaded model and abbrev to match appropriate SHAP explainer'''

    '''Must always use training dataset as background data in order to
        evaluate SHAP values for either testing (usually)or training set'''

    explainer = None
    trained_model = model


    if abbrev in ["NB"]:
        explainer = shap.Explainer(trained_model.predict, trainX)


        # dont use model.predict for Linear Explainer (only for Explainer)
        # ^^^ You get a class method error when creating shap plots and values
    if abbrev in ["LR"]:
        explainer = shap.LinearExplainer(trained_model, trainX)


    if abbrev in ['DT', 'RF', "XGB", "LGB","CGB"]:
        explainer = shap.TreeExplainer(trained_model)



    return explainer
```

SHAP: compute_shapValues()

NOTES

- Parameter 'X' in this context refers to whatever training or testing dataset that was passed in from the whole run from below
- Mentioned earlier, default run uses training dataset as background data and creates shap values using testing data

- The same follows for feature_names --> either train_feat or test_feat (default) will be passed

In [10]:

```
def compute_shapValues(model, abbrev, explainer, X) -> pd.DataFrame :

    '''This method will calculate shapley values and store these as a Pandas DataFrame for conversion to csv file
       This includes creating expected_values and shap_values --> returns shap_values (will be called by shap_summary)
    '''

    max_evals = max(500, (2 * len(X)) + 1)   # optional: declares number of permutations for shap.Explainer()
    shap_values = None

    if abbrev in ["NB"]:
        shap_values= explainer(X)  # permutation object cannot use .expected_value function like LR
        print(shap_values)



    if abbrev in ["LR"]:
        shap_values = explainer.shap_values(X)
        print(shap_values)


    # i think shap_values() only works for TreeExplainer and LinearExplainer...Explainer for NB is considered a
    # permutation object
    if abbrev in ['DT', 'RF', "XGB", "LGB","CGB"]:
        shap_values = explainer.shap_values(X, approximate=False, check_additivity=False)
        print(shap_values)



    return shap_values
```

SHAP: shap_summary()

Plot Types for SHAP v0.41.0

Waterfall

- Plots an explantion of a single prediction as a waterfall plot

Summary (type: violin & bar)

- Summary plots of SHAP values across a whole dataset

Dependence

- Plots the value of the feature on the x-axis and the SHAP value of the same feature on the y-axis
- This shows how the model depends on the given feature, and is like a richer extenstion of the classical parital dependence plots.
- Vertical dispersion of the data points represents interaction effects.
- Grey ticks along the y-axis are data points where the feature's value was NaN.

Force

- Visualize cumulative SHAP values with an additive force layout.

Beeswarm

- Summary plots of SHAP values across a whole dataset
- Designed to display an information-dense summary of how the top features in a dataset impact the model's output.

In [11]:

```
def shap_summary(abbrev, feature_names,shap_values, explainer, X, cvCount):
    '''Retrieve shap_values from previous method;
            this method will return and display different types of shap plots
    '''

    save_path = experiment_path + '/hcc-data_example/model_evaluation/shap_values/testResults/shapFigures/'
    # checks algorithm in given list to execute shap summaries
    if abbrev in ["NB"]:

        print('Summary Plot for SHAP Values in Class 0 & 1 in Test Set: \n')
        shap.summary_plot(shap_values, X, feature_names, plot_type='violin', show=False)
            # print('SHAP Bar Plot for Summary Plot for SHAP Values in Class 0 & 1 in Test Set:\n')
#         shap.plots.bar(shap_values.values) # doesnt work but should for this...attribute error

        print('SHAP Beeswarm Plot for Top 5 SHAP Values in Class 0 & 1 in Test Set: \n')
        shap.plots.beeswarm(shap_values, max_display=5, show=False)  #max_display allows user to choose # of features

            # print('Waterfall Plot for SHAP Values in Class 0 in Test Set: \n')
#         shap.waterfall_plot(shap_values, max_display=5, show=True)  # should work for this model
```

```
    # #     # scatter, bar, waterfall, beeswarm plots should work for this model
    # #     # waterfall plot also doesnt work...i get "AttributeError: 'numpy.ndarray' object has no attribute 'base_v
    # #       Bar plot should work for this model if using .Explainer() and shap_values = explainer(data)-->
    # #       not explainer.shap_values
    elif abbrev in ["LR", 'XGB']:

        expected_value = explainer.expected_value
        print('Expected value for {}: {}'.format(abbrev, expected_value))

        print('Summary Plot for SHAP Values in Test Set: \n')
        shap.summary_plot(shap_values, X, feature_names, plot_type='violin', show=False)
        plt.savefig(save_path+abbrev+'_'+str(cvCount)+'shapSummaryPlot.pdf')
        plt.close()

        print('SHAP Bar Plot for SHAP Values Test Set: \n')
        shap.summary_plot(shap_values, X, feature_names, plot_type="bar", show=False)
        plt.savefig(save_path+abbrev+'_'+str(cvCount)+'shapSummaryBarPlot.pdf')
        plt.close()


        print('SHAP Decision Plot for SHAP Values in Test Set: \n')
        shap.decision_plot(expected_value, shap_values, feature_names, show=False)
        plt.savefig(save_path+abbrev+'_'+str(cvCount)+'shapDecisionPlot.pdf')
        plt.close()


        print('SHAP Decision Plot for Single-Prediction in Test Set: \n')
        shap.decision_plot(expected_value, shap_values[54], feature_names, show=False)
        plt.savefig(save_path+abbrev+'_'+str(cvCount)+'shapDecisionPlot_singlePredict.pdf')
        plt.close()



            # waterfall plot works for DT() if it uses .Explainer() and shap_vales = explainer(data)
            # instead of using TreeExplainer but other plots listed here work
    elif abbrev in ['DT', 'RF', 'LGB','CGB']:
        expected_value = explainer.expected_value
        print('Expected value for {}: {}'.format(abbrev, expected_value))

        print('Bar Summary Plot for SHAP Values in Class 0 & 1 in Test Set: \n')
            #       #tree.tree_plot(testX)  ---> helps display Decision Tree
        shap.summary_plot(shap_values, X, feature_names, plot_type='bar', class_names=['0', '1'], show=False)


        print('\nDecision Plot for SHAP Values from Class 0 in Test Set: \n')
        shap.decision_plot(expected_value[0], shap_values[0], feature_names=feature_names, show=False)

        print('\nDecision Plot for SHAP Values from Class 1 in Test Set: \n')
        shap.decision_plot(expected_value[1], shap_values[1], feature_names=feature_names, show=False)

```

In [12]:

```
# def run_force_plot(model, abbrev, explainer, shap_values, trainX, testX, run = True):
#     if abbrev in ['NB']:
#         print('\nForce Plot for {} SHAP Values from Class 0 in Test Set: \n'.format(abbrev))
#         shap.force_plot(shap_values[0], testX.iloc[0], feature_names=feature_names, show=True)

#         print('\nForce Plot for {} SHAP Values from Class 0 in Test Set: \n'.format(abbrev))
#         shap.force_plot(shap_values[1], testX.iloc[1], feature_names=feature_names, show=True)


#     elif abbrev in ['LR', 'XGB']:
#         print('\nChecking if shap plots are returned and consistent...\n')
#         summary = shap_summary(algorithms, shap_values, explainer, trainX, testX)  # retrieve shap summary plots

#         print('\nForce Plot for SHAP Values in  Whole Test Set:  \n')
#         shap.force_plot(explainer.expected_value, shap_values, testX)


#     else:
#         print('\nForce Plot for {} SHAP Values from Class 0 in Test Set: \n'.format(abbrev))
#         shap.force_plot(explainer.expected_value[0], shap_values[0], feature_names=feature_names)

#         print('\nForce Plot for {} SHAP Values from Class 1 in Test Set: \n'.format(abbrev))
#         shap.force_plot(explainer.expected_value[1], shap_values[1], feature_names=feature_names)
```

^^^ fix later ... may want to keep this to create force plots and save results

In [13]:

```
def shap_feature_ranking(abbrev, shap_values, X, feature_names):  # 'X' and 'feature_names' argument is whichever test

    '''Calculate the average of the absolute SHAP values for each feature and use it to show
       which features were the most important when making a prediction'''

    if abbrev in ['NB']:
        feature_order = np.argsort(np.mean(np.abs(shap_values.values), axis=0))
        df = pd.DataFrame({"Features": [feature_names[i] for i in feature_order][::-1],"Importance": [ np.mean(np.abs(

    elif abbrev in ['LR','LGB', 'XGB', 'CGB']: #LR cant use shap_values.values
        feature_order = np.argsort(np.mean(np.abs(shap_values), axis=0))
        df = pd.DataFrame({"Features": [feature_names[i] for i in feature_order][::-1],"Importance": [ np.mean(np.abs(
```

```python

    else: # For multiclass models (can be used for NB)..Loops through Class 0 and Class 1
         #  Sums up the shap average values form both classes to get the shap average for the whole CV for the model

        c_idxs = []
        columns = feature_names
        for column in range(0, (len(columns))):
            if isinstance(shap_values, list):
                c_idxs.append(X.columns.get_loc(column))
                means = [np.abs(shap_values[class_][:, c_idxs]).mean(axis=0) for class_ in range(len(shap_values))]
                shap_means = np.sum(np.column_stack(means), 1)
            else:                                    # Else there is only one 2D array of shap values
                assert len(shap_values.shape) == 2, 'Expected two-dimensional shap values array.'
                shap_means = np.abs(shap_values).mean(axis=0)
        df = pd.DataFrame({'Features': feature_names, 'Importance': shap_means}).sort_values(by='Importance', ascendin
        df.index += 1

    return df
```

In [14]:

```python
def merge_shap_df(df) -> pd.DataFrame: # 'df' parameter is the dataframe returned from shap_feature_ranking()
    # create a new dataframe that stores the model's shap feature importance values over each CV

    result = pd.DataFrame()
    result = pd.concat(df, join='outer', axis=1).fillna('void')
    display(result)

#     new_df = pd.DataFrame()
#     new_df = pd.merge(new_df, df)

    return result
```

Next Steps

- Saving shap figures per model in each cv
- Make sure you can loop through each pickled model, load it, create shap values and display plots
- Be able to load one model at a time, create shapley values for each CV train and test set, store shap scores in a dataframe
- Make sure to load original dataset features so that each csv file is the same length as the original dataset
    - This means when a CV dataset is missing a feature, we make sure to assign a shap score of 0
    - each new csv file for loading shap scores of each trained model must include all features

```
LR_shap_all_CVs.csv ==>
                        LR_0 --> CV0
                        LR_1 --> CV1
                        LR_2 --> CV2
```

- Save dataframe for each model in a csv file

More Updates/Fixes

8/02/22

- Currently unsure if creating DataFrame for each model's shap_values should be done in compute_shap_values() or within the nested for-loop in testing cell

8/04/22

- Can create DataFrames for each CV but feature names most likely are not matching actual values (double check it)
- Difficult looping through to merge Dataframes for all CVs features...tried temporary variable
- Must also consider that shap_values array are returned in order of features from test/train set it was passed from...not based on feature order in test/train set **FIXED on 8/05/22**
    - Consider mapping out and ordering the values to avoid shuffling of names and values **FIXED on 8/05/22**

8/05/22

- Saving feature importance scores for each cv
- Created two different runs, one for actual test (default) and another if the user chooses to run it on the training sets for comparison

8/08/22

- Iterating through multiclass shap values for Decision Tree poses issue?...ideally we'd want to get the shap absolute average for both classes 0 and 1...same might be for XGB and any other model that has multiclass output **FIXED on 8/08/22**
    - Figured out that when running the loop in shap_feature_ranking() for Decision Tree, both classes 0 and 1 are accounted for. The shap absolute averages are summed up automatically to get the overall CV feature importances for the model (i double checked this myself through creating a loop that would output two different csv files for each class it iterated through)

- **Current issue:** Figuring out how to save multiple figures for each model when calling shap_summary()...for now, I can only save each figure individually through each CV...if model NB has 2 plot function calls & iterate through 3 CVs --> total 6 shap plots for **ONE** model .....

Run SHAP for Testing Datasets

Loop through each hcc_demo dataset to unpickle and load trained models to create Shapley values and plots Default run

- The default setting runs explainer and shap values for the TESTING datasets for each model and CV
- User has the option below to run the loop for training sets as well

In [18]:

```
# testing all methods
run_force_plots = False # parameter in run_force_plot(); set to True if user wants to display force plots for trained
run_test = True
save_path = experiment_path + '/hcc-data_example/model_evaluation/shap_values/testResults/shapFigures/'


if run_test == True:
    for each in datasets:
        print("----------------------------------------")
        print(each)
        print("----------------------------------------")
        full_path = experiment_path+'/' + each

        #Make folder in experiment folder/datafolder to store all shap_values per algorithm/CV combination
        if not os.path.exists(full_path+'/model_evaluation/shap_values/testResults'):
            os.mkdir(full_path+'/model_evaluation/shap_values/testResults')

        original_headers = pd.read_csv(full_path+"/exploratory/OriginalFeatureNames.csv",sep=',').columns.values.tolis
        feat_order_map = {feat:i for i, feat in enumerate(original_headers)}
        print(feat_order_map)

        for algorithm in algorithms: #loop through algorithms
            print(abbrev[algorithm])


            for cvCount in range(0,cv_partitions): #loop through cv's
                print('{}{} In CV{}...'.format(abbrev[algorithm], cvCount, cvCount))

                # unpickle and load model
                result_file = full_path+ '/models/pickledModels/' + abbrev[algorithm]+ "_" + str(cvCount)+".pickle"
                file = open(result_file, 'rb')
                model = pickle.load(file)
                file.close()
                print('\nChecking if correct model is loaded...\n', model)


                # Load CV datasets, paths to datasets updates with each iteration
                train_path = experiment_path +  '/' + each + '/CVDatasets/'+  '/'+ each + '_CV_' + str(cvCount) + '_Tr
                test_path = experiment_path +  '/' + each+  '/CVDatasets/'+ '/'+ each +'_CV_' + str(cvCount) + '_Test.
                trainX, trainY,testX, testY, train_feat, test_feat = dataPrep(train_path,instance_label,class_label, t


                # shap computation and plots
                # Sanity check: print explainer to check if explainer exists
                explainer = get_explainer(model, abbrev[algorithm], trainX)  #explainer must always use training set
                print('\nChecking explainer for {}{}...\n{}'.format(abbrev[algorithm], cvCount, explainer))

                print('\nChecking shap values for {}{}...\n'.format(abbrev[algorithm], cvCount))
                shap_values = compute_shapValues(model, abbrev[algorithm], explainer, testX)

                print('\nChecking shap plots for {}{}...\n'.format(abbrev[algorithm], cvCount))
                shap_summary(abbrev[algorithm], test_feat, shap_values, explainer, testX, cvCount)
#                 plt.savefig(full_path + "/model_evaluation/shap_values/testResults/shapFigures/shapPlots_" + abbrev[
#                 plt.close()
#                 plt.clf()
#                 fig = plt.figure()
#                 plt.savefig(fig)
# #                 plt.savefig(save_path + abbrev[algorithm] + '_' + str(cvCount)+'_shapPlots.pdf',  bbox_inches='tig
#                 plt.close()



                #save SHAP FI results
                print('\nChecking feature importance for {}{}...\n'.format(abbrev[algorithm], cvCount))
                shap_fi_df = shap_feature_ranking(abbrev[algorithm], shap_values, testX, test_feat) # can either choos
                filepath = full_path+"/model_evaluation/shap_values/testResults/"+ abbrev[algorithm] + '_' + str(cvCou
                shap_fi_df.to_csv(filepath, header=True, index=True)

                     # create new folder to save summary plots for each model per CV
#                 if not os.path.exists(experiment_path+'/hcc-data_example/model_evaluation/shap_values/testResults/sh
#                     os.mkdir(full_path+'/model_evaluation/shap_values/testResults/shapFigures')

#                     filepath2 = full_path+"/model_evaluation/shap_values/testResults/shapFigures"+ abbrev[algorithm]
#                     summary.to_pdf(filepath2, header=True, index=True)

                # only runs force plots if run = True
                if run_force_plots == True:
                    if abbrev[algorithm] in ['NB']:

                        print('\nForce Plot for {}{} SHAP Values in Test Set: \n'.format(abbrev[algorithm], cvCount))
                        shap.force_plot(shap_values, feature_names = test_feat)

                        print('\nSingle-Prediction Force Plot for {}{} SHAP Values in Test Set: \n'.format(abbrev[algo
                        shap.force_plot(shap_values[42], testX.iloc[42], feature_names=test_feat, show=True)
                        break


#                     elif abbrev[algorithm] in ['LR', 'XGB', 'LGB', 'CBG']:  #need to test out LGB and CBG for this

#                         print('\nForce Plot for {}{} SHAP Values in Whole Test Set:  \n'.format(abbrev[algorithm], c
#                         shap.force_plot(explainer.expected_value, shap_values, testX, feature_names=test_feat)

#                         print('\nSingle-Prediction Force Plot for {}{} SHAP Values in Test Set: \n'.format(abbrev[al
#                         shap.force_plot(explainer.expected_value, shap_values[42], testX.iloc[42], feature_names=tes
#                         break


#                     else:
#                        # Decision Tree has multiclass output so needed to create two separate function calls
#                        # Decision Tree doesn't work when just using shap_values as a parameter
#                         print('\nForce Plot for {}{} SHAP Values from Class 0 in Test Set: \n'.format(abbrev[algorit
#                         shap.force_plot(explainer.expected_value[0], shap_values[0], feature_names=test_feat)

#                         print('\nForce Plot for {}{} SHAP Values from Class 1 in Test Set: \n'.format(abbrev[algorit
#                         shap.force_plot(explainer.expected_value[1], shap_values[1], feature_names=test_feat)
#                         break
```

```
--------------------------------------
hcc-data_example
--------------------------------------
{'Gender': 0, 'Symptoms ': 1, 'Alcohol': 2, 'Hepatitis B Surface Antigen': 3, 'Hepatitis B e Antigen': 4, 'Hepatitis B
Core Antibody': 5, 'Hepatitis C Virus Antibody': 6, 'Cirrhosis': 7, 'Endemic Countries': 8, 'Smoking': 9, 'Diabetes':
10, 'Obesity': 11, 'Hemochromatosis': 12, 'Arterial Hypertension': 13, 'Chronic Renal Insufficiency': 14, 'Human Immun
odeficiency Virus': 15, 'Nonalcoholic Steatohepatitis': 16, 'Esophageal Varices': 17, 'Splenomegaly': 18, 'Portal Hype
rtension': 19, 'Portal Vein Thrombosis': 20, 'Liver Metastasis': 21, 'Radiological Hallmark': 22, 'Age at diagnosis':
23, 'Grams of Alcohol per day': 24, 'Packs of cigarets per year': 25, 'Performance Status*': 26, 'Encephalopathy degre
e*': 27, 'Ascites degree*': 28, 'International Normalised Ratio*': 29, 'Alpha-Fetoprotein (ng/mL)': 30, 'Haemoglobin
(g/dL)': 31, 'Mean Corpuscular Volume': 32, 'Leukocytes(G/L)': 33, 'Platelets': 34, 'Albumin (mg/dL)': 35, 'Total Bili
rubin(mg/dL)': 36, 'Alanine transaminase (U/L)': 37, 'Aspartate transaminase (U/L)': 38, 'Gamma glutamyl transferase
(U/L)': 39, 'Alkaline phosphatase (U/L)': 40, 'Total Proteins (g/dL)': 41, 'Creatinine (mg/dL)': 42, 'Number of Nodule
s': 43, 'Major dimension of nodule (cm)': 44, 'Direct Bilirubin (mg/dL)': 45, 'Iron': 46, 'Oxygen Saturation (%)': 47,
'Ferritin (ng/mL)': 48}
NB
NB0 In CV0...

Checking if correct model is loaded...
 GaussianNB()
['Alanine transaminase (U/L)', 'Albumin (mg/dL)', 'Alkaline phosphatase (U/L)', 'Alpha-Fetoprotein (ng/mL)', 'Arterial
Hypertension', 'Ascites degree*', 'Aspartate transaminase (U/L)', 'Chronic Renal Insufficiency', 'Cirrhosis', 'Creatin
ine (mg/dL)', 'Diabetes', 'Direct Bilirubin (mg/dL)', 'Encephalopathy degree*', 'Endemic Countries', 'Esophageal Varic
es', 'Ferritin (ng/mL)', 'Gamma glutamyl transferase (U/L)', 'Haemoglobin (g/dL)', 'Hemochromatosis', 'Hepatitis B Sur
face Antigen', 'Hepatitis B e Antigen', 'Hepatitis C Virus Antibody', 'International Normalised Ratio*', 'Iron', 'Leuk
ocytes(G/L)', 'Liver Metastasis', 'Major dimension of nodule (cm)', 'Mean Corpuscular Volume', 'Number of Nodules', 'O
besity', 'Oxygen Saturation (%)', 'Packs of cigarets per year', 'Performance Status*', 'Portal Hypertension', 'Portal
Vein Thrombosis', 'Smoking', 'Splenomegaly', 'Symptoms ', 'Total Bilirubin(mg/dL)']
['Alanine transaminase (U/L)', 'Albumin (mg/dL)', 'Alkaline phosphatase (U/L)', 'Alpha-Fetoprotein (ng/mL)', 'Arterial
Hypertension', 'Ascites degree*', 'Aspartate transaminase (U/L)', 'Chronic Renal Insufficiency', 'Cirrhosis', 'Creatin
ine (mg/dL)', 'Diabetes', 'Direct Bilirubin (mg/dL)', 'Encephalopathy degree*', 'Endemic Countries', 'Esophageal Varic
es', 'Ferritin (ng/mL)', 'Gamma glutamyl transferase (U/L)', 'Haemoglobin (g/dL)', 'Hemochromatosis', 'Hepatitis B Sur
face Antigen', 'Hepatitis B e Antigen', 'Hepatitis C Virus Antibody', 'International Normalised Ratio*', 'Iron', 'Leuk
ocytes(G/L)', 'Liver Metastasis', 'Major dimension of nodule (cm)', 'Mean Corpuscular Volume', 'Number of Nodules', 'O
besity', 'Oxygen Saturation (%)', 'Packs of cigarets per year', 'Performance Status*', 'Portal Hypertension', 'Portal
Vein Thrombosis', 'Smoking', 'Splenomegaly', 'Symptoms ', 'Total Bilirubin(mg/dL)']

Checking explainer for NB0...
shap.explainers.Permutation()

Checking shap values for NB0...

.values =
array([[ 0.00333333, -0.035     , -0.01583333, ..., -0.00083333,
         0.005     , -0.02083333],
       [ 0.01      ,  0.01833333,  0.05166667, ...,  0.00416667,
         0.02166667, -0.02333333],
       [ 0.01083333, -0.00333333, -0.00166667, ...,  0.00666667,
         0.04333333, -0.01333333],
       ...,
       [-0.00166667, -0.00666667, -0.00916667, ...,  0.        ,
         0.00333333, -0.02583333],
       [-0.00583333,  0.00083333, -0.01583333, ...,  0.        ,
        -0.02083333, -0.02666667],
       [ 0.01166667,  0.0425    , -0.00916667, ...,  0.00416667,
        -0.1       ,  0.43833333]])

.base_values =
array([0.33, 0.33, 0.33, 0.33, 0.33, 0.33, 0.33, 0.33, 0.33, 0.33, 0.33,
       0.33, 0.33, 0.33, 0.33, 0.33, 0.33, 0.33, 0.33, 0.33, 0.33, 0.33,
       0.33, 0.33, 0.33, 0.33, 0.33, 0.33, 0.33, 0.33, 0.33, 0.33, 0.33,
       0.33, 0.33, 0.33, 0.33, 0.33, 0.33, 0.33, 0.33, 0.33, 0.33, 0.33,
       0.33, 0.33, 0.33, 0.33, 0.33, 0.33, 0.33, 0.33, 0.33, 0.33, 0.33])

.data =
array([[ 0.0368995,  1.0551286, -0.5852099, ..., -1.2247449,  0.6264224,
        -0.4383745],
       [-0.3323658, -0.2719725,  3.8874108, ...,  0.8164966,  0.6264224,
        -0.3900995],
       [-0.0973788,  0.3178502,  0.1974987, ...,  0.8164966,  0.6264224,
        -0.4866495],
       ...,
       [-0.6680615,  0.6127615, -0.708207 , ...,  0.8164966,  0.6264224,
        -0.3740079],
       [-0.9198333,  0.3178502, -0.484576 , ...,  0.8164966, -1.5963668,
        -0.2613663],
       [ 0.1376082, -0.3457004, -0.3615789, ...,  0.8164966, -1.5963668,
         1.1546993]])

Checking shap plots for NB0...

Summary Plot for SHAP Values in Class 0 & 1 in Test Set:

SHAP Beeswarm Plot for Top 5 SHAP Values in Class 0 & 1 in Test Set:


Checking feature importance for NB0...

NB1 In CV1...

Checking if correct model is loaded...
 GaussianNB()
['Age at diagnosis', 'Alanine transaminase (U/L)', 'Albumin (mg/dL)', 'Alcohol', 'Alkaline phosphatase (U/L)', 'Alpha-
Fetoprotein (ng/mL)', 'Ascites degree*', 'Aspartate transaminase (U/L)', 'Cirrhosis', 'Creatinine (mg/dL)', 'Diabete
s', 'Direct Bilirubin (mg/dL)', 'Endemic Countries', 'Esophageal Varices', 'Ferritin (ng/mL)', 'Gamma glutamyl transfe
rase (U/L)', 'Grams of Alcohol per day', 'Haemoglobin (g/dL)', 'Hepatitis B Core Antibody', 'Hepatitis B Surface Antig
en', 'Hepatitis B e Antigen', 'Hepatitis C Virus Antibody', 'International Normalised Ratio*', 'Iron', 'Leukocytes(G/
L)', 'Liver Metastasis', 'Major dimension of nodule (cm)', 'Mean Corpuscular Volume', 'Nonalcoholic Steatohepatitis',
'Number of Nodules', 'Obesity', 'Oxygen Saturation (%)', 'Packs of cigarets per year', 'Performance Status*', 'Platele
ts', 'Portal Hypertension', 'Portal Vein Thrombosis', 'Smoking', 'Symptoms ', 'Total Bilirubin(mg/dL)', 'Total Protein
s (g/dL)']
['Age at diagnosis', 'Alanine transaminase (U/L)', 'Albumin (mg/dL)', 'Alcohol', 'Alkaline phosphatase (U/L)', 'Alpha-
Fetoprotein (ng/mL)', 'Ascites degree*', 'Aspartate transaminase (U/L)', 'Cirrhosis', 'Creatinine (mg/dL)', 'Diabete
s', 'Direct Bilirubin (mg/dL)', 'Endemic Countries', 'Esophageal Varices', 'Ferritin (ng/mL)', 'Gamma glutamyl transfe
rase (U/L)', 'Grams of Alcohol per day', 'Haemoglobin (g/dL)', 'Hepatitis B Core Antibody', 'Hepatitis B Surface Antig
en', 'Hepatitis B e Antigen', 'Hepatitis C Virus Antibody', 'International Normalised Ratio*', 'Iron', 'Leukocytes(G/
L)', 'Liver Metastasis', 'Major dimension of nodule (cm)', 'Mean Corpuscular Volume', 'Nonalcoholic Steatohepatitis',
'Number of Nodules', 'Obesity', 'Oxygen Saturation (%)', 'Packs of cigarets per year', 'Performance Status*', 'Platele
ts', 'Portal Hypertension', 'Portal Vein Thrombosis', 'Smoking', 'Symptoms ', 'Total Bilirubin(mg/dL)', 'Total Protein
s (g/dL)']

Checking explainer for NB1...
shap.explainers.Permutation()

Checking shap values for NB1...

.values =
array([[-0.0225    ,  0.02166667, -0.09166667, ...,  0.1125    ,
        -0.01416667, -0.01      ],
       [ 0.0375    , -0.00833333, -0.03      , ..., -0.1825    ,
        -0.01833333, -0.02083333],
       [ 0.02833333,  0.01166667,  0.045     , ..., -0.0875    ,
        -0.01      , -0.00666667],
       ...,
       [-0.025     , -0.01      ,  0.01916667, ..., -0.18416667,
        -0.01083333, -0.01083333],
       [ 0.00083333,  0.00083333,  0.00166667, ...,  0.00916667,
        -0.005     , -0.00333333],
       [ 0.04083333,  0.01416667, -0.0575    , ...,  0.085     ,
        -0.01666667, -0.01      ]])

.base_values =
array([0.49, 0.49, 0.49, 0.49, 0.49, 0.49, 0.49, 0.49, 0.49, 0.49, 0.49,
       0.49, 0.49, 0.49, 0.49, 0.49, 0.49, 0.49, 0.49, 0.49, 0.49, 0.49,
       0.49, 0.49, 0.49, 0.49, 0.49, 0.49, 0.49, 0.49, 0.49, 0.49, 0.49,
       0.49, 0.49, 0.49, 0.49, 0.49, 0.49, 0.49, 0.49, 0.49, 0.49, 0.49,
       0.49, 0.49, 0.49, 0.49, 0.49, 0.49, 0.49, 0.49, 0.49, 0.49, 0.49])

.data =
array([[-0.7178553, -0.3317865,  1.9018142, ...,  0.6831301, -0.4188214,
        -0.0949777],
       [ 0.9544964, -0.9389338,  0.0448948, ..., -1.4638501, -0.4505005,
        -0.2135745],
       [ 0.5909417, -0.2169208, -0.558604 , ..., -1.4638501, -0.1812288,
        -0.176513 ],
       ...,
       [-0.4270115,  0.83328  ,  0.1996381, ..., -1.4638501, -0.3713029,
        -0.2358114],
       [-0.4270115,  1.1450584, -0.1098485, ...,  0.6831301, -0.2445869,
        -0.2358114],
       [ 1.0272073, -0.6763836,  1.1280978, ...,  0.6831301, -0.3871424,
        -0.2580483]])

Checking shap plots for NB1...

Summary Plot for SHAP Values in Class 0 & 1 in Test Set:

SHAP Beeswarm Plot for Top 5 SHAP Values in Class 0 & 1 in Test Set:


Checking feature importance for NB1...

NB2 In CV2...

Checking if correct model is loaded...
 GaussianNB()
['Age at diagnosis', 'Albumin (mg/dL)', 'Alcohol', 'Alkaline phosphatase (U/L)', 'Alpha-Fetoprotein (ng/mL)', 'Arteria
l Hypertension', 'Ascites degree*', 'Aspartate transaminase (U/L)', 'Chronic Renal Insufficiency', 'Creatinine (mg/d
L)', 'Diabetes', 'Direct Bilirubin (mg/dL)', 'Encephalopathy degree*', 'Endemic Countries', 'Ferritin (ng/mL)', 'Gamma
glutamyl transferase (U/L)', 'Haemoglobin (g/dL)', 'Hepatitis B Core Antibody', 'Hepatitis B e Antigen', 'Hepatitis C
Virus Antibody', 'Human Immunodeficiency Virus', 'International Normalised Ratio*', 'Iron', 'Leukocytes(G/L)', 'Liver
Metastasis', 'Major dimension of nodule (cm)', 'Mean Corpuscular Volume', 'Number of Nodules', 'Obesity', 'Oxygen Satu
ration (%)', 'Packs of cigarets per year', 'Performance Status*', 'Platelets', 'Portal Hypertension', 'Portal Vein Thr
ombosis', 'Symptoms ']
['Age at diagnosis', 'Albumin (mg/dL)', 'Alcohol', 'Alkaline phosphatase (U/L)', 'Alpha-Fetoprotein (ng/mL)', 'Arteria
l Hypertension', 'Ascites degree*', 'Aspartate transaminase (U/L)', 'Chronic Renal Insufficiency', 'Creatinine (mg/d
L)', 'Diabetes', 'Direct Bilirubin (mg/dL)', 'Encephalopathy degree*', 'Endemic Countries', 'Ferritin (ng/mL)', 'Gamma
glutamyl transferase (U/L)', 'Haemoglobin (g/dL)', 'Hepatitis B Core Antibody', 'Hepatitis B e Antigen', 'Hepatitis C
Virus Antibody', 'Human Immunodeficiency Virus', 'International Normalised Ratio*', 'Iron', 'Leukocytes(G/L)', 'Liver
Metastasis', 'Major dimension of nodule (cm)', 'Mean Corpuscular Volume', 'Number of Nodules', 'Obesity', 'Oxygen Satu
ration (%)', 'Packs of cigarets per year', 'Performance Status*', 'Platelets', 'Portal Hypertension', 'Portal Vein Thr
ombosis', 'Symptoms ']

Checking explainer for NB2...
shap.explainers.Permutation()
```

```
Checking shap values for NB2...

.values =
array([[ 0.        ,  0.        ,  0.        , ...,  0.        ,
         0.        ,  0.        ],
       [ 0.005     ,  0.        ,  0.        , ...,  0.005     ,
         0.        ,  0.        ],
       [-0.00166667,  0.0075    , -0.02666667, ..., -0.0025    ,
        -0.00833333,  0.02083333],
       ...,
       [-0.00416667,  0.03416667,  0.        , ..., -0.00166667,
        -0.00666667,  0.0025    ],
       [-0.00583333, -0.04      ,  0.005     , ..., -0.00583333,
        -0.015     ,  0.0175    ],
       [ 0.        ,  0.00166667,  0.        , ...,  0.        ,
         0.        ,  0.        ]])

.base_values =
array([0.05, 0.05, 0.05, 0.05, 0.05, 0.05, 0.05, 0.05, 0.05, 0.05, 0.05,
       0.05, 0.05, 0.05, 0.05, 0.05, 0.05, 0.05, 0.05, 0.05, 0.05, 0.05,
       0.05, 0.05, 0.05, 0.05, 0.05, 0.05, 0.05, 0.05, 0.05, 0.05, 0.05,
       0.05, 0.05, 0.05, 0.05, 0.05, 0.05, 0.05, 0.05, 0.05, 0.05, 0.05,
       0.05, 0.05, 0.05, 0.05, 0.05, 0.05, 0.05, 0.05, 0.05, 0.05, 0.05])

.data =
array([[-0.8437697, -0.4184704,  0.5703518, ...,  0.6264224, -0.5282705,
        -1.3228757],
       [-3.7262129, -0.4184704,  0.5703518, ..., -1.5963668, -0.5282705,
         0.7559289],
       [-1.0908363,  0.0205566, -1.7533038, ...,  0.6264224, -0.5282705,
         0.7559289],
       ...,
       [-1.0908363, -1.8818937,  0.5703518, ...,  0.6264224, -0.5282705,
         0.7559289],
       [-1.2555473,  1.0449529,  0.5703518, ...,  0.6264224, -0.5282705,
         0.7559289],
       [-0.1025701, -2.028236 ,  0.5703518, ...,  0.6264224, -0.5282705,
         0.7559289]])

Checking shap plots for NB2...

Summary Plot for SHAP Values in Class 0 & 1 in Test Set:

SHAP Beeswarm Plot for Top 5 SHAP Values in Class 0 & 1 in Test Set:


Checking feature importance for NB2...

LR
LR0 In CV0...

Checking if correct model is loaded...
 LogisticRegression(C=0.006606805070193189, dual=True,
                   max_iter=193.8544995971634, random_state=42,
                   solver='liblinear')
['Alanine transaminase (U/L)', 'Albumin (mg/dL)', 'Alkaline phosphatase (U/L)', 'Alpha-Fetoprotein (ng/mL)', 'Arterial
Hypertension', 'Ascites degree*', 'Aspartate transaminase (U/L)', 'Chronic Renal Insufficiency', 'Cirrhosis', 'Creatin
ine (mg/dL)', 'Diabetes', 'Direct Bilirubin (mg/dL)', 'Encephalopathy degree*', 'Endemic Countries', 'Esophageal Varic
es', 'Ferritin (ng/mL)', 'Gamma glutamyl transferase (U/L)', 'Haemoglobin (g/dL)', 'Hemochromatosis', 'Hepatitis B Sur
face Antigen', 'Hepatitis B e Antigen', 'Hepatitis C Virus Antibody', 'International Normalised Ratio*', 'Iron', 'Leuk
ocytes(G/L)', 'Liver Metastasis', 'Major dimension of nodule (cm)', 'Mean Corpuscular Volume', 'Number of Nodules', 'O
besity', 'Oxygen Saturation (%)', 'Packs of cigarets per year', 'Performance Status*', 'Portal Hypertension', 'Portal
Vein Thrombosis', 'Smoking', 'Splenomegaly', 'Symptoms ', 'Total Bilirubin(mg/dL)']
['Alanine transaminase (U/L)', 'Albumin (mg/dL)', 'Alkaline phosphatase (U/L)', 'Alpha-Fetoprotein (ng/mL)', 'Arterial
Hypertension', 'Ascites degree*', 'Aspartate transaminase (U/L)', 'Chronic Renal Insufficiency', 'Cirrhosis', 'Creatin
ine (mg/dL)', 'Diabetes', 'Direct Bilirubin (mg/dL)', 'Encephalopathy degree*', 'Endemic Countries', 'Esophageal Varic
es', 'Ferritin (ng/mL)', 'Gamma glutamyl transferase (U/L)', 'Haemoglobin (g/dL)', 'Hemochromatosis', 'Hepatitis B Sur
face Antigen', 'Hepatitis B e Antigen', 'Hepatitis C Virus Antibody', 'International Normalised Ratio*', 'Iron', 'Leuk
ocytes(G/L)', 'Liver Metastasis', 'Major dimension of nodule (cm)', 'Mean Corpuscular Volume', 'Number of Nodules', 'O
besity', 'Oxygen Saturation (%)', 'Packs of cigarets per year', 'Performance Status*', 'Portal Hypertension', 'Portal
Vein Thrombosis', 'Smoking', 'Splenomegaly', 'Symptoms ', 'Total Bilirubin(mg/dL)']

Checking explainer for LR0...
<shap.explainers._linear.Linear object at 0x7f9bb26ea9a0>

Checking shap values for LR0...

[[-1.54247265e-04 -5.20008470e-02 -4.40485958e-02 ... -1.47969848e-03
   2.31157863e-02 -2.02239904e-02]
 [ 6.85778818e-04  1.94894821e-02  2.13041189e-01 ...  1.21066239e-03
   2.31157863e-02 -1.80842415e-02]
 [ 1.51216786e-04 -1.22839969e-02  9.42115197e-04 ...  1.21066239e-03
   2.31157863e-02 -2.23637392e-02]
 ...
 [ 1.44943883e-03 -2.81707337e-02 -5.11185666e-02 ...  1.21066239e-03
   2.31157863e-02 -1.73709948e-02]
 [ 2.02218390e-03 -1.22839969e-02 -3.82640794e-02 ...  1.21066239e-03
  -6.24982370e-02 -1.23782505e-02]
 [-3.83345247e-04  2.34611703e-02 -3.11941086e-02 ...  1.21066239e-03
  -6.24982370e-02  5.03876684e-02]]

Checking shap plots for LR0...
```

```
Expected value for LR: -0.023696555525940875
Summary Plot for SHAP Values in Test Set:

SHAP Bar Plot for SHAP Values Test Set:

SHAP Decision Plot for SHAP Values in Test Set:

SHAP Decision Plot for Single-Prediction in Test Set:


Checking feature importance for LR0...

LR1 In CV1...

Checking if correct model is loaded...
 LogisticRegression(C=0.06359900885943309, max_iter=48.076782938152924,
                   random_state=42, solver='sag')
['Age at diagnosis', 'Alanine transaminase (U/L)', 'Albumin (mg/dL)', 'Alcohol', 'Alkaline phosphatase (U/L)', 'Alpha-
Fetoprotein (ng/mL)', 'Ascites degree*', 'Aspartate transaminase (U/L)', 'Cirrhosis', 'Creatinine (mg/dL)', 'Diabete
s', 'Direct Bilirubin (mg/dL)', 'Endemic Countries', 'Esophageal Varices', 'Ferritin (ng/mL)', 'Gamma glutamyl transfe
rase (U/L)', 'Grams of Alcohol per day', 'Haemoglobin (g/dL)', 'Hepatitis B Core Antibody', 'Hepatitis B Surface Antig
en', 'Hepatitis B e Antigen', 'Hepatitis C Virus Antibody', 'International Normalised Ratio*', 'Iron', 'Leukocytes(G/
L)', 'Liver Metastasis', 'Major dimension of nodule (cm)', 'Mean Corpuscular Volume', 'Nonalcoholic Steatohepatitis',
'Number of Nodules', 'Obesity', 'Oxygen Saturation (%)', 'Packs of cigarets per year', 'Performance Status*', 'Platele
ts', 'Portal Hypertension', 'Portal Vein Thrombosis', 'Smoking', 'Symptoms ', 'Total Bilirubin(mg/dL)', 'Total Protein
s (g/dL)']
['Age at diagnosis', 'Alanine transaminase (U/L)', 'Albumin (mg/dL)', 'Alcohol', 'Alkaline phosphatase (U/L)', 'Alpha-
Fetoprotein (ng/mL)', 'Ascites degree*', 'Aspartate transaminase (U/L)', 'Cirrhosis', 'Creatinine (mg/dL)', 'Diabete
s', 'Direct Bilirubin (mg/dL)', 'Endemic Countries', 'Esophageal Varices', 'Ferritin (ng/mL)', 'Gamma glutamyl transfe
rase (U/L)', 'Grams of Alcohol per day', 'Haemoglobin (g/dL)', 'Hepatitis B Core Antibody', 'Hepatitis B Surface Antig
en', 'Hepatitis B e Antigen', 'Hepatitis C Virus Antibody', 'International Normalised Ratio*', 'Iron', 'Leukocytes(G/
L)', 'Liver Metastasis', 'Major dimension of nodule (cm)', 'Mean Corpuscular Volume', 'Nonalcoholic Steatohepatitis',
'Number of Nodules', 'Obesity', 'Oxygen Saturation (%)', 'Packs of cigarets per year', 'Performance Status*', 'Platele
ts', 'Portal Hypertension', 'Portal Vein Thrombosis', 'Smoking', 'Symptoms ', 'Total Bilirubin(mg/dL)', 'Total Protein
s (g/dL)']

Checking explainer for LR1...
<shap.explainers._linear.Linear object at 0x7f9bb3930400>

Checking shap values for LR1...

[[-0.19905492  0.03850256 -0.30345684 ...  0.14567922 -0.09989041
  -0.01025168]
 [ 0.18762272  0.10977384 -0.00291614 ... -0.39387344 -0.10644632
  -0.02397595]
 [ 0.10356236  0.0250188   0.09475959 ... -0.39387344 -0.05072119
  -0.01968712]
 ...
 [-0.13180663 -0.09826126 -0.0279612  ... -0.39387344 -0.09005657
  -0.02654925]
 [-0.13180663 -0.13486004  0.02212892 ...  0.14567922 -0.063833
  -0.02654925]
 [ 0.20443478  0.07895383 -0.17823155 ...  0.14567922 -0.09333451
  -0.02912255]]

Checking shap plots for LR1...

Expected value for LR: -0.6091565598361125
Summary Plot for SHAP Values in Test Set:

SHAP Bar Plot for SHAP Values Test Set:

SHAP Decision Plot for SHAP Values in Test Set:

SHAP Decision Plot for Single-Prediction in Test Set:


Checking feature importance for LR1...

LR2 In CV2...

Checking if correct model is loaded...
 LogisticRegression(C=0.0006580360277501316, class_weight='balanced', dual=True,
                   max_iter=112.07606211860569, random_state=42,
                   solver='liblinear')
['Age at diagnosis', 'Albumin (mg/dL)', 'Alcohol', 'Alkaline phosphatase (U/L)', 'Alpha-Fetoprotein (ng/mL)', 'Arteria
l Hypertension', 'Ascites degree*', 'Aspartate transaminase (U/L)', 'Chronic Renal Insufficiency', 'Creatinine (mg/d
L)', 'Diabetes', 'Direct Bilirubin (mg/dL)', 'Encephalopathy degree*', 'Endemic Countries', 'Ferritin (ng/mL)', 'Gamma
glutamyl transferase (U/L)', 'Haemoglobin (g/dL)', 'Hepatitis B Core Antibody', 'Hepatitis B e Antigen', 'Hepatitis C
Virus Antibody', 'Human Immunodeficiency Virus', 'International Normalised Ratio*', 'Iron', 'Leukocytes(G/L)', 'Liver
Metastasis', 'Major dimension of nodule (cm)', 'Mean Corpuscular Volume', 'Number of Nodules', 'Obesity', 'Oxygen Satu
ration (%)', 'Packs of cigarets per year', 'Performance Status*', 'Platelets', 'Portal Hypertension', 'Portal Vein Thr
ombosis', 'Symptoms ']
['Age at diagnosis', 'Albumin (mg/dL)', 'Alcohol', 'Alkaline phosphatase (U/L)', 'Alpha-Fetoprotein (ng/mL)', 'Arteria
l Hypertension', 'Ascites degree*', 'Aspartate transaminase (U/L)', 'Chronic Renal Insufficiency', 'Creatinine (mg/d
L)', 'Diabetes', 'Direct Bilirubin (mg/dL)', 'Encephalopathy degree*', 'Endemic Countries', 'Ferritin (ng/mL)', 'Gamma
glutamyl transferase (U/L)', 'Haemoglobin (g/dL)', 'Hepatitis B Core Antibody', 'Hepatitis B e Antigen', 'Hepatitis C
Virus Antibody', 'Human Immunodeficiency Virus', 'International Normalised Ratio*', 'Iron', 'Leukocytes(G/L)', 'Liver
Metastasis', 'Major dimension of nodule (cm)', 'Mean Corpuscular Volume', 'Number of Nodules', 'Obesity', 'Oxygen Satu
ration (%)', 'Packs of cigarets per year', 'Performance Status*', 'Platelets', 'Portal Hypertension', 'Portal Vein Thr
ombosis', 'Symptoms ']

Checking explainer for LR2...
```

```
<shap.explainers._linear.Linear object at 0x7f9bb39ff6a0>

Checking shap values for LR2...

[[-0.00209083  0.00438031  0.00118267 ... -0.00203892 -0.00439287
  -0.00869176]
 [-0.00982644  0.00438031  0.00118267 ...  0.00524293 -0.00439287
   0.00355015]
 [-0.00275388  0.00044561 -0.00473069 ... -0.00203892 -0.00439287
   0.00355015]
 ...
 [-0.00275388  0.01749597  0.00118267 ... -0.00203892 -0.00439287
   0.00355015]
 [-0.00319591 -0.00873536  0.00118267 ... -0.00203892 -0.00439287
   0.00355015]
 [-0.00010167  0.01880754  0.00118267 ... -0.00203892 -0.00439287
   0.00355015]]

Checking shap plots for LR2...

Expected value for LR: -0.006133751932115765
Summary Plot for SHAP Values in Test Set:

SHAP Bar Plot for SHAP Values Test Set:

SHAP Decision Plot for SHAP Values in Test Set:

SHAP Decision Plot for Single-Prediction in Test Set:


Checking feature importance for LR2...

DT
DT0 In CV0...

Checking if correct model is loaded...
 DecisionTreeClassifier(max_depth=17, min_samples_leaf=35, min_samples_split=45,
                       random_state=42)
['Alanine transaminase (U/L)', 'Albumin (mg/dL)', 'Alkaline phosphatase (U/L)', 'Alpha-Fetoprotein (ng/mL)', 'Arterial
Hypertension', 'Ascites degree*', 'Aspartate transaminase (U/L)', 'Chronic Renal Insufficiency', 'Cirrhosis', 'Creatin
ine (mg/dL)', 'Diabetes', 'Direct Bilirubin (mg/dL)', 'Encephalopathy degree*', 'Endemic Countries', 'Esophageal Varic
es', 'Ferritin (ng/mL)', 'Gamma glutamyl transferase (U/L)', 'Haemoglobin (g/dL)', 'Hemochromatosis', 'Hepatitis B Sur
face Antigen', 'Hepatitis B e Antigen', 'Hepatitis C Virus Antibody', 'International Normalised Ratio*', 'Iron', 'Leuk
ocytes(G/L)', 'Liver Metastasis', 'Major dimension of nodule (cm)', 'Mean Corpuscular Volume', 'Number of Nodules', 'O
besity', 'Oxygen Saturation (%)', 'Packs of cigarets per year', 'Performance Status*', 'Portal Hypertension', 'Portal
Vein Thrombosis', 'Smoking', 'Splenomegaly', 'Symptoms ', 'Total Bilirubin(mg/dL)']
['Alanine transaminase (U/L)', 'Albumin (mg/dL)', 'Alkaline phosphatase (U/L)', 'Alpha-Fetoprotein (ng/mL)', 'Arterial
Hypertension', 'Ascites degree*', 'Aspartate transaminase (U/L)', 'Chronic Renal Insufficiency', 'Cirrhosis', 'Creatin
ine (mg/dL)', 'Diabetes', 'Direct Bilirubin (mg/dL)', 'Encephalopathy degree*', 'Endemic Countries', 'Esophageal Varic
es', 'Ferritin (ng/mL)', 'Gamma glutamyl transferase (U/L)', 'Haemoglobin (g/dL)', 'Hemochromatosis', 'Hepatitis B Sur
face Antigen', 'Hepatitis B e Antigen', 'Hepatitis C Virus Antibody', 'International Normalised Ratio*', 'Iron', 'Leuk
ocytes(G/L)', 'Liver Metastasis', 'Major dimension of nodule (cm)', 'Mean Corpuscular Volume', 'Number of Nodules', 'O
besity', 'Oxygen Saturation (%)', 'Packs of cigarets per year', 'Performance Status*', 'Portal Hypertension', 'Portal
Vein Thrombosis', 'Smoking', 'Splenomegaly', 'Symptoms ', 'Total Bilirubin(mg/dL)']

Checking explainer for DT0...
<shap.explainers._tree.Tree object at 0x7f9bb587e220>

Checking shap values for DT0...

[array([[0., 0., 0., ..., 0., 0., 0.],
       [0., 0., 0., ..., 0., 0., 0.],
       [0., 0., 0., ..., 0., 0., 0.],
       ...,
       [0., 0., 0., ..., 0., 0., 0.],
       [0., 0., 0., ..., 0., 0., 0.],
       [0., 0., 0., ..., 0., 0., 0.]]), array([[0., 0., 0., ..., 0., 0., 0.],
       [0., 0., 0., ..., 0., 0., 0.],
       [0., 0., 0., ..., 0., 0., 0.],
       ...,
       [0., 0., 0., ..., 0., 0., 0.],
       [0., 0., 0., ..., 0., 0., 0.],
       [0., 0., 0., ..., 0., 0., 0.]])]

Checking shap plots for DT0...

Expected value for DT: [0.57272727 0.42727273]
Bar Summary Plot for SHAP Values in Class 0 & 1 in Test Set:


Decision Plot for SHAP Values from Class 0 in Test Set:


Decision Plot for SHAP Values from Class 1 in Test Set:


Checking feature importance for DT0...

DT1 In CV1...

Checking if correct model is loaded...
 DecisionTreeClassifier(criterion='entropy', max_depth=21, min_samples_leaf=3,
                       min_samples_split=23, random_state=42,
                        splitter='random')
['Age at diagnosis', 'Alanine transaminase (U/L)', 'Albumin (mg/dL)', 'Alcohol', 'Alkaline phosphatase (U/L)', 'Alpha-
Fetoprotein (ng/mL)', 'Ascites degree*', 'Aspartate transaminase (U/L)', 'Cirrhosis', 'Creatinine (mg/dL)', 'Diabete
s', 'Direct Bilirubin (mg/dL)', 'Endemic Countries', 'Esophageal Varices', 'Ferritin (ng/mL)', 'Gamma glutamyl transfe
rase (U/L)', 'Grams of Alcohol per day', 'Haemoglobin (g/dL)', 'Hepatitis B Core Antibody', 'Hepatitis B Surface Antig
en', 'Hepatitis B e Antigen', 'Hepatitis C Virus Antibody', 'International Normalised Ratio*', 'Iron', 'Leukocytes(G/
L)', 'Liver Metastasis', 'Major dimension of nodule (cm)', 'Mean Corpuscular Volume', 'Nonalcoholic Steatohepatitis',
'Number of Nodules', 'Obesity', 'Oxygen Saturation (%)', 'Packs of cigarets per year', 'Performance Status*', 'Platele
ts', 'Portal Hypertension', 'Portal Vein Thrombosis', 'Smoking', 'Symptoms ', 'Total Bilirubin(mg/dL)', 'Total Protein
s (g/dL)']
['Age at diagnosis', 'Alanine transaminase (U/L)', 'Albumin (mg/dL)', 'Alcohol', 'Alkaline phosphatase (U/L)', 'Alpha-
Fetoprotein (ng/mL)', 'Ascites degree*', 'Aspartate transaminase (U/L)', 'Cirrhosis', 'Creatinine (mg/dL)', 'Diabete
s', 'Direct Bilirubin (mg/dL)', 'Endemic Countries', 'Esophageal Varices', 'Ferritin (ng/mL)', 'Gamma glutamyl transfe
rase (U/L)', 'Grams of Alcohol per day', 'Haemoglobin (g/dL)', 'Hepatitis B Core Antibody', 'Hepatitis B Surface Antig
en', 'Hepatitis B e Antigen', 'Hepatitis C Virus Antibody', 'International Normalised Ratio*', 'Iron', 'Leukocytes(G/
L)', 'Liver Metastasis', 'Major dimension of nodule (cm)', 'Mean Corpuscular Volume', 'Nonalcoholic Steatohepatitis',
'Number of Nodules', 'Obesity', 'Oxygen Saturation (%)', 'Packs of cigarets per year', 'Performance Status*', 'Platele
ts', 'Portal Hypertension', 'Portal Vein Thrombosis', 'Smoking', 'Symptoms ', 'Total Bilirubin(mg/dL)', 'Total Protein
s (g/dL)']

Checking explainer for DT1...
<shap.explainers._tree.Tree object at 0x7f9bb5866ee0>

Checking shap values for DT1...

[array([[ 0.        ,  0.        ,  0.        , ..., -0.02795699,
          0.        ,  0.        ],
        [ 0.        ,  0.        ,  0.        , ...,  0.07004662,
          0.        ,  0.        ],
        [ 0.        ,  0.        ,  0.        , ...,  0.07004662,
          0.        ,  0.        ],
        ...,
        [ 0.        ,  0.        ,  0.        , ...,  0.08666667,
          0.        ,  0.        ],
        [ 0.        ,  0.        ,  0.        , ..., -0.11345397,
          0.        ,  0.        ],
        [ 0.        ,  0.        ,  0.        , ..., -0.02795699,
          0.        ,  0.        ]]), array([[ 0.        ,  0.        ,  0.        , ...,  0.02795699,
          0.        ,  0.        ],
        [ 0.        ,  0.        ,  0.        , ..., -0.07004662,
          0.        ,  0.        ],
        [ 0.        ,  0.        ,  0.        , ..., -0.07004662,
          0.        ,  0.        ],
        ...,
        [ 0.        ,  0.        ,  0.        , ..., -0.08666667,
          0.        ,  0.        ],
        [ 0.        ,  0.        ,  0.        , ...,  0.11345397,
          0.        ,  0.        ],
        [ 0.        ,  0.        ,  0.        , ...,  0.02795699,
          0.        ,  0.        ]])]

Checking shap plots for DT1...

Expected value for DT: [0.63636364 0.36363636]
Bar Summary Plot for SHAP Values in Class 0 & 1 in Test Set:


Decision Plot for SHAP Values from Class 0 in Test Set:


Decision Plot for SHAP Values from Class 1 in Test Set:


Checking feature importance for DT1...

DT2 In CV2...

Checking if correct model is loaded...
 DecisionTreeClassifier(class_weight='balanced', max_depth=29,
                        min_samples_leaf=30, min_samples_split=45,
                        random_state=42)
['Age at diagnosis', 'Albumin (mg/dL)', 'Alcohol', 'Alkaline phosphatase (U/L)', 'Alpha-Fetoprotein (ng/mL)', 'Arteria
l Hypertension', 'Ascites degree*', 'Aspartate transaminase (U/L)', 'Chronic Renal Insufficiency', 'Creatinine (mg/d
L)', 'Diabetes', 'Direct Bilirubin (mg/dL)', 'Encephalopathy degree*', 'Endemic Countries', 'Ferritin (ng/mL)', 'Gamma
glutamyl transferase (U/L)', 'Haemoglobin (g/dL)', 'Hepatitis B Core Antibody', 'Hepatitis B e Antigen', 'Hepatitis C
Virus Antibody', 'Human Immunodeficiency Virus', 'International Normalised Ratio*', 'Iron', 'Leukocytes(G/L)', 'Liver
Metastasis', 'Major dimension of nodule (cm)', 'Mean Corpuscular Volume', 'Number of Nodules', 'Obesity', 'Oxygen Satu
ration (%)', 'Packs of cigarets per year', 'Performance Status*', 'Platelets', 'Portal Hypertension', 'Portal Vein Thr
ombosis', 'Symptoms ']
['Age at diagnosis', 'Albumin (mg/dL)', 'Alcohol', 'Alkaline phosphatase (U/L)', 'Alpha-Fetoprotein (ng/mL)', 'Arteria
l Hypertension', 'Ascites degree*', 'Aspartate transaminase (U/L)', 'Chronic Renal Insufficiency', 'Creatinine (mg/d
L)', 'Diabetes', 'Direct Bilirubin (mg/dL)', 'Encephalopathy degree*', 'Endemic Countries', 'Ferritin (ng/mL)', 'Gamma
glutamyl transferase (U/L)', 'Haemoglobin (g/dL)', 'Hepatitis B Core Antibody', 'Hepatitis B e Antigen', 'Hepatitis C
Virus Antibody', 'Human Immunodeficiency Virus', 'International Normalised Ratio*', 'Iron', 'Leukocytes(G/L)', 'Liver
Metastasis', 'Major dimension of nodule (cm)', 'Mean Corpuscular Volume', 'Number of Nodules', 'Obesity', 'Oxygen Satu
ration (%)', 'Packs of cigarets per year', 'Performance Status*', 'Platelets', 'Portal Hypertension', 'Portal Vein Thr
ombosis', 'Symptoms ']

Checking explainer for DT2...
<shap.explainers._tree.Tree object at 0x7f9b70aca070>

Checking shap values for DT2...
```

```
[array([[ 0.        , -0.14385676,  0.        , ...,  0.        ,
          0.        ,  0.        ],
        [ 0.        , -0.0462963 ,  0.        , ...,  0.        ,
          0.        ,  0.        ],
        [ 0.        , -0.0462963 ,  0.        , ...,  0.        ,
          0.        ,  0.        ],
        ...,
        [ 0.        , -0.0462963 ,  0.        , ...,  0.        ,
          0.        ,  0.        ],
        [ 0.        ,  0.05769231,  0.        , ...,  0.        ,
          0.        ,  0.        ],
        [ 0.        , -0.14385676,  0.        , ...,  0.        ,
          0.        ,  0.        ]]), array([[ 0.        ,  0.14385676,  0.        , ...,  0.        ,
          0.        ,  0.        ],
        [ 0.        ,  0.0462963 ,  0.        , ...,  0.        ,
          0.        ,  0.        ],
        [ 0.        ,  0.0462963 ,  0.        , ...,  0.        ,
          0.        ,  0.        ],
        ...,
        [ 0.        ,  0.0462963 ,  0.        , ...,  0.        ,
          0.        ,  0.        ],
        [ 0.        , -0.05769231,  0.        , ...,  0.        ,
          0.        ,  0.        ],
        [ 0.        ,  0.14385676,  0.        , ...,  0.        ,
          0.        ,  0.        ]])]

Checking shap plots for DT2...

Expected value for DT: [0.5 0.5]
Bar Summary Plot for SHAP Values in Class 0 & 1 in Test Set:


Decision Plot for SHAP Values from Class 0 in Test Set:


Decision Plot for SHAP Values from Class 1 in Test Set:


Checking feature importance for DT2...

RF
RF0 In CV0...

Checking if correct model is loaded...
 RandomForestClassifier(criterion='entropy', max_depth=1, max_features=None,
                      min_samples_leaf=17, min_samples_split=41,
                      n_estimators=960, random_state=42)
['Alanine transaminase (U/L)', 'Albumin (mg/dL)', 'Alkaline phosphatase (U/L)', 'Alpha-Fetoprotein (ng/mL)', 'Arterial
Hypertension', 'Ascites degree*', 'Aspartate transaminase (U/L)', 'Chronic Renal Insufficiency', 'Cirrhosis', 'Creatin
ine (mg/dL)', 'Diabetes', 'Direct Bilirubin (mg/dL)', 'Encephalopathy degree*', 'Endemic Countries', 'Esophageal Varic
es', 'Ferritin (ng/mL)', 'Gamma glutamyl transferase (U/L)', 'Haemoglobin (g/dL)', 'Hemochromatosis', 'Hepatitis B Sur
face Antigen', 'Hepatitis B e Antigen', 'Hepatitis C Virus Antibody', 'International Normalised Ratio*', 'Iron', 'Leuk
ocytes(G/L)', 'Liver Metastasis', 'Major dimension of nodule (cm)', 'Mean Corpuscular Volume', 'Number of Nodules', 'O
besity', 'Oxygen Saturation (%)', 'Packs of cigarets per year', 'Performance Status*', 'Portal Hypertension', 'Portal
Vein Thrombosis', 'Smoking', 'Splenomegaly', 'Symptoms ', 'Total Bilirubin(mg/dL)']
['Alanine transaminase (U/L)', 'Albumin (mg/dL)', 'Alkaline phosphatase (U/L)', 'Alpha-Fetoprotein (ng/mL)', 'Arterial
Hypertension', 'Ascites degree*', 'Aspartate transaminase (U/L)', 'Chronic Renal Insufficiency', 'Cirrhosis', 'Creatin
ine (mg/dL)', 'Diabetes', 'Direct Bilirubin (mg/dL)', 'Encephalopathy degree*', 'Endemic Countries', 'Esophageal Varic
es', 'Ferritin (ng/mL)', 'Gamma glutamyl transferase (U/L)', 'Haemoglobin (g/dL)', 'Hemochromatosis', 'Hepatitis B Sur
face Antigen', 'Hepatitis B e Antigen', 'Hepatitis C Virus Antibody', 'International Normalised Ratio*', 'Iron', 'Leuk
ocytes(G/L)', 'Liver Metastasis', 'Major dimension of nodule (cm)', 'Mean Corpuscular Volume', 'Number of Nodules', 'O
besity', 'Oxygen Saturation (%)', 'Packs of cigarets per year', 'Performance Status*', 'Portal Hypertension', 'Portal
Vein Thrombosis', 'Smoking', 'Splenomegaly', 'Symptoms ', 'Total Bilirubin(mg/dL)']

Checking explainer for RF0...
<shap.explainers._tree.Tree object at 0x7f9b70c37610>

Checking shap values for RF0...

[array([[ 0.        ,  0.04181896,  0.00365806, ...,  0.        ,
          0.        ,  0.        ],
        [ 0.        , -0.01898534, -0.01999966, ...,  0.        ,
          0.        ,  0.        ],
        [ 0.        , -0.01898534, -0.01840236, ...,  0.        ,
          0.        ,  0.        ],
        ...,
        [ 0.        ,  0.04012049,  0.02614831, ...,  0.        ,
          0.        ,  0.        ],
        [ 0.        , -0.01898534, -0.00631506, ...,  0.        ,
          0.        ,  0.        ],
        [ 0.        , -0.01898534, -0.01061794, ...,  0.        ,
          0.        ,  0.        ]]), array([[ 0.        , -0.04181896, -0.00365806, ...,  0.        ,
          0.        ,  0.        ],
        [ 0.        ,  0.01898534,  0.01999966, ...,  0.        ,
          0.        ,  0.        ],
        [ 0.        ,  0.01898534,  0.01840236, ...,  0.        ,
          0.        ,  0.        ],
        ...,
        [ 0.        , -0.04012049, -0.02614831, ...,  0.        ,
          0.        ,  0.        ],
        [ 0.        ,  0.01898534,  0.00631506, ...,  0.        ,
          0.        ,  0.        ],
        [ 0.        ,  0.01898534,  0.01061794, ...,  0.        ,
          0.        ,  0.        ]]])]

 Checking shap plots for RF0...

 Expected value for RF: [0.5728125 0.4271875]
 Bar Summary Plot for SHAP Values in Class 0 & 1 in Test Set:


 Decision Plot for SHAP Values from Class 0 in Test Set:


 Decision Plot for SHAP Values from Class 1 in Test Set:


 Checking feature importance for RF0...

 RF1 In CV1...

 Checking if correct model is loaded...
  RandomForestClassifier(class_weight='balanced', criterion='entropy',
                        max_depth=2, max_features='log2', min_samples_leaf=9,
                        min_samples_split=31, n_estimators=207, random_state=42)
 ['Age at diagnosis', 'Alanine transaminase (U/L)', 'Albumin (mg/dL)', 'Alcohol', 'Alkaline phosphatase (U/L)', 'Alpha-
 Fetoprotein (ng/mL)', 'Ascites degree*', 'Aspartate transaminase (U/L)', 'Cirrhosis', 'Creatinine (mg/dL)', 'Diabete
 s', 'Direct Bilirubin (mg/dL)', 'Endemic Countries', 'Esophageal Varices', 'Ferritin (ng/mL)', 'Gamma glutamyl transfe
 rase (U/L)', 'Grams of Alcohol per day', 'Haemoglobin (g/dL)', 'Hepatitis B Core Antibody', 'Hepatitis B Surface Antig
 en', 'Hepatitis B e Antigen', 'Hepatitis C Virus Antibody', 'International Normalised Ratio*', 'Iron', 'Leukocytes(G/
 L)', 'Liver Metastasis', 'Major dimension of nodule (cm)', 'Mean Corpuscular Volume', 'Nonalcoholic Steatohepatitis',
 'Number of Nodules', 'Obesity', 'Oxygen Saturation (%)', 'Packs of cigarets per year', 'Performance Status*', 'Platele
 ts', 'Portal Hypertension', 'Portal Vein Thrombosis', 'Smoking', 'Symptoms ', 'Total Bilirubin(mg/dL)', 'Total Protein
 s (g/dL)']
 ['Age at diagnosis', 'Alanine transaminase (U/L)', 'Albumin (mg/dL)', 'Alcohol', 'Alkaline phosphatase (U/L)', 'Alpha-
 Fetoprotein (ng/mL)', 'Ascites degree*', 'Aspartate transaminase (U/L)', 'Cirrhosis', 'Creatinine (mg/dL)', 'Diabete
 s', 'Direct Bilirubin (mg/dL)', 'Endemic Countries', 'Esophageal Varices', 'Ferritin (ng/mL)', 'Gamma glutamyl transfe
 rase (U/L)', 'Grams of Alcohol per day', 'Haemoglobin (g/dL)', 'Hepatitis B Core Antibody', 'Hepatitis B Surface Antig
 en', 'Hepatitis B e Antigen', 'Hepatitis C Virus Antibody', 'International Normalised Ratio*', 'Iron', 'Leukocytes(G/
 L)', 'Liver Metastasis', 'Major dimension of nodule (cm)', 'Mean Corpuscular Volume', 'Nonalcoholic Steatohepatitis',
 'Number of Nodules', 'Obesity', 'Oxygen Saturation (%)', 'Packs of cigarets per year', 'Performance Status*', 'Platele
 ts', 'Portal Hypertension', 'Portal Vein Thrombosis', 'Smoking', 'Symptoms ', 'Total Bilirubin(mg/dL)', 'Total Protein
 s (g/dL)']

 Checking explainer for RF1...
 <shap.explainers._tree.Tree object at 0x7f9bb6912d30>

 Checking shap values for RF1...

 [array([[ 3.95998286e-03,  8.78166228e-05,  5.42231605e-02, ...,
         -4.99207302e-03,  7.28955367e-03,  2.80322166e-03],
        [-3.19678857e-03,  8.79064012e-04,  1.60776918e-03, ...,
          1.80742612e-02,  5.37725417e-03, -1.63881455e-03],
        [-2.91886671e-03, -4.37616095e-04, -2.66173097e-02, ...,
          2.10142415e-02,  3.35299303e-03,  5.93880632e-04],
        ...,
        [ 2.62833254e-03, -1.78372567e-04,  1.24812736e-02, ...,
          1.73079030e-02,  4.94635866e-03, -1.53940799e-03],
        [ 2.33387141e-03, -4.42763000e-04, -2.58700195e-02, ...,
         -6.86380765e-03,  4.07652440e-03, -1.52713434e-03],
        [-5.01472471e-03,  1.13411715e-03,  5.63221558e-02, ...,
         -5.51702530e-03,  5.22884231e-03, -4.18441541e-04]]), array([[-3.95998286e-03, -8.78166228e-05, -5.42231605e-0
 2, ...,
          4.99207302e-03, -7.28955367e-03, -2.80322166e-03],
        [ 3.19678857e-03, -8.79064012e-04, -1.60776918e-03, ...,
         -1.80742612e-02, -5.37725417e-03,  1.63881455e-03],
        [ 2.91886671e-03,  4.37616095e-04,  2.66173097e-02, ...,
         -2.10142415e-02, -3.35299303e-03, -5.93880632e-04],
        ...,
        [-2.62833254e-03,  1.78372567e-04, -1.24812736e-02, ...,
         -1.73079030e-02, -4.94635866e-03,  1.53940799e-03],
        [-2.33387141e-03,  4.42763000e-04,  2.58700195e-02, ...,
          6.86380765e-03, -4.07652440e-03,  1.52713434e-03],
        [ 5.01472471e-03, -1.13411715e-03, -5.63221558e-02, ...,
          5.51702530e-03, -5.22884231e-03,  4.18441541e-04]])]

 Checking shap plots for RF1...

 Expected value for RF: [0.49673858 0.50326142]
 Bar Summary Plot for SHAP Values in Class 0 & 1 in Test Set:


 Decision Plot for SHAP Values from Class 0 in Test Set:


 Decision Plot for SHAP Values from Class 1 in Test Set:


 Checking feature importance for RF1...

 RF2 In CV2...

 Checking if correct model is loaded...
  RandomForestClassifier(max_depth=11, max_features=None, min_samples_leaf=14,
                        min_samples_split=27, n_estimators=10, random_state=42)
 ['Age at diagnosis', 'Albumin (mg/dL)', 'Alcohol', 'Alkaline phosphatase (U/L)', 'Alpha-Fetoprotein (ng/mL)', 'Arteria
```

```
l Hypertension', 'Ascites degree*', 'Aspartate transaminase (U/L)', 'Chronic Renal Insufficiency', 'Creatinine (mg/d
L)', 'Diabetes', 'Direct Bilirubin (mg/dL)', 'Encephalopathy degree*', 'Endemic Countries', 'Ferritin (ng/mL)', 'Gamma
glutamyl transferase (U/L)', 'Haemoglobin (g/dL)', 'Hepatitis B Core Antibody', 'Hepatitis B e Antigen', 'Hepatitis C
Virus Antibody', 'Human Immunodeficiency Virus', 'International Normalised Ratio*', 'Iron', 'Leukocytes(G/L)', 'Liver
Metastasis', 'Major dimension of nodule (cm)', 'Mean Corpuscular Volume', 'Number of Nodules', 'Obesity', 'Oxygen Satu
ration (%)', 'Packs of cigarets per year', 'Performance Status*', 'Platelets', 'Portal Hypertension', 'Portal Vein Thr
ombosis', 'Symptoms ']
['Age at diagnosis', 'Albumin (mg/dL)', 'Alcohol', 'Alkaline phosphatase (U/L)', 'Alpha-Fetoprotein (ng/mL)', 'Arteria
l Hypertension', 'Ascites degree*', 'Aspartate transaminase (U/L)', 'Chronic Renal Insufficiency', 'Creatinine (mg/d
L)', 'Diabetes', 'Direct Bilirubin (mg/dL)', 'Encephalopathy degree*', 'Endemic Countries', 'Ferritin (ng/mL)', 'Gamma
glutamyl transferase (U/L)', 'Haemoglobin (g/dL)', 'Hepatitis B Core Antibody', 'Hepatitis B e Antigen', 'Hepatitis C
Virus Antibody', 'Human Immunodeficiency Virus', 'International Normalised Ratio*', 'Iron', 'Leukocytes(G/L)', 'Liver
Metastasis', 'Major dimension of nodule (cm)', 'Mean Corpuscular Volume', 'Number of Nodules', 'Obesity', 'Oxygen Satu
ration (%)', 'Packs of cigarets per year', 'Performance Status*', 'Platelets', 'Portal Hypertension', 'Portal Vein Thr
ombosis', 'Symptoms ']

Checking explainer for RF2...
<shap.explainers._tree.Tree object at 0x7f9bb691e370>

Checking shap values for RF2...

[array([[ 0.01794027, -0.01378879,  0.        , ...,  0.        ,
          0.        ,  0.        ],
        [ 0.00711062, -0.00448347,  0.        , ...,  0.        ,
          0.        ,  0.        ],
        [ 0.00711062,  0.00318182,  0.        , ...,  0.        ,
          0.        ,  0.        ],
        ...,
        [ 0.00711062, -0.00448347,  0.        , ...,  0.        ,
          0.        ,  0.        ],
        [ 0.01210694,  0.00318182,  0.        , ...,  0.        ,
          0.        ,  0.        ],
        [ 0.01794027, -0.01378879,  0.        , ...,  0.        ,
          0.        ,  0.        ]]), array([[-0.01794027,  0.01378879,  0.        , ...,  0.        ,
          0.        ,  0.        ],
        [-0.00711062,  0.00448347,  0.        , ...,  0.        ,
          0.        ,  0.        ],
        [-0.00711062, -0.00318182,  0.        , ...,  0.        ,
          0.        ,  0.        ],
        ...,
        [-0.00711062,  0.00448347,  0.        , ...,  0.        ,
          0.        ,  0.        ],
        [-0.01210694, -0.00318182,  0.        , ...,  0.        ,
          0.        ,  0.        ],
        [-0.01794027,  0.01378879,  0.        , ...,  0.        ,
          0.        ,  0.        ]])]

Checking shap plots for RF2...

Expected value for RF: [0.61909091 0.38090909]
Bar Summary Plot for SHAP Values in Class 0 & 1 in Test Set:


Decision Plot for SHAP Values from Class 0 in Test Set:


Decision Plot for SHAP Values from Class 1 in Test Set:


Checking feature importance for RF2...

XGB
XGB0 In CV0...

Checking if correct model is loaded...
 XGBClassifier(alpha=0.0002575842389979265, base_score=0.5, booster='gbtree',
              callbacks=None, colsample_bylevel=1, colsample_bynode=1,
              colsample_bytree=0.9181376162919086, early_stopping_rounds=None,
              enable_categorical=False, eta=5.623331491160975e-07,
              eval_metric=None, gamma=0.0002786718840103683, gpu_id=-1,
              grow_policy='lossguide', importance_type=None,
              interaction_constraints='', learning_rate=5.62333128e-07,
              max_bin=256, max_cat_to_onehot=4, max_delta_step=0, max_depth=27,
              max_leaves=0, min_child_weight=0.20525460238584922,
              min_samples_leaf=27, min_samples_split=37, missing=nan,
              monotone_constraints='()', n_estimators=164, n_jobs=1, nthread=1, ...)
['Alanine transaminase (U/L)', 'Albumin (mg/dL)', 'Alkaline phosphatase (U/L)', 'Alpha-Fetoprotein (ng/mL)', 'Arterial
Hypertension', 'Ascites degree*', 'Aspartate transaminase (U/L)', 'Chronic Renal Insufficiency', 'Cirrhosis', 'Creatin
ine (mg/dL)', 'Diabetes', 'Direct Bilirubin (mg/dL)', 'Encephalopathy degree*', 'Endemic Countries', 'Esophageal Varic
es', 'Ferritin (ng/mL)', 'Gamma glutamyl transferase (U/L)', 'Haemoglobin (g/dL)', 'Hemochromatosis', 'Hepatitis B Sur
face Antigen', 'Hepatitis B e Antigen', 'Hepatitis C Virus Antibody', 'International Normalised Ratio*', 'Iron', 'Leuk
ocytes(G/L)', 'Liver Metastasis', 'Major dimension of nodule (cm)', 'Mean Corpuscular Volume', 'Number of Nodules', 'O
besity', 'Oxygen Saturation (%)', 'Packs of cigarets per year', 'Performance Status*', 'Portal Hypertension', 'Portal
Vein Thrombosis', 'Smoking', 'Splenomegaly', 'Symptoms ', 'Total Bilirubin(mg/dL)']
['Alanine transaminase (U/L)', 'Albumin (mg/dL)', 'Alkaline phosphatase (U/L)', 'Alpha-Fetoprotein (ng/mL)', 'Arterial
Hypertension', 'Ascites degree*', 'Aspartate transaminase (U/L)', 'Chronic Renal Insufficiency', 'Cirrhosis', 'Creatin
ine (mg/dL)', 'Diabetes', 'Direct Bilirubin (mg/dL)', 'Encephalopathy degree*', 'Endemic Countries', 'Esophageal Varic
es', 'Ferritin (ng/mL)', 'Gamma glutamyl transferase (U/L)', 'Haemoglobin (g/dL)', 'Hemochromatosis', 'Hepatitis B Sur
face Antigen', 'Hepatitis B e Antigen', 'Hepatitis C Virus Antibody', 'International Normalised Ratio*', 'Iron', 'Leuk
ocytes(G/L)', 'Liver Metastasis', 'Major dimension of nodule (cm)', 'Mean Corpuscular Volume', 'Number of Nodules', 'O
besity', 'Oxygen Saturation (%)', 'Packs of cigarets per year', 'Performance Status*', 'Portal Hypertension', 'Portal
Vein Thrombosis', 'Smoking', 'Splenomegaly', 'Symptoms ', 'Total Bilirubin(mg/dL)']
```

```
Checking explainer for XGB0...
<shap.explainers._tree.Tree object at 0x7f9b70fb6880>

Checking shap values for XGB0...

[[-3.0952360e-06 -3.7619997e-05 -2.3696571e-06 ... -1.2713954e-07
   1.3318062e-07 -2.2155659e-06]
 [-1.1003718e-05  1.4350392e-05  2.1088497e-06 ...  6.4105130e-08
   4.7997673e-07  2.5433717e-06]
 [-3.2040905e-06  1.4657915e-05  1.6045622e-05 ...  2.1734811e-08
   1.6514244e-07 -8.6456166e-06]
 ...
 [ 1.4764141e-06 -2.2817851e-05 -1.9707142e-05 ...  4.7425072e-08
   1.5751922e-07  6.8362704e-07]
 [ 1.7882336e-05  9.4924808e-06  9.1785603e-07 ...  2.1734811e-08
  -1.7996480e-06  7.2638748e-09]
 [-3.0676049e-06  1.4631669e-05  5.5639280e-06 ...  2.1734811e-08
  -1.6272652e-06  2.0639069e-07]]

Checking shap plots for XGB0...

Expected value for XGB: 1.0591810450932826e-06
Summary Plot for SHAP Values in Test Set:

SHAP Bar Plot for SHAP Values Test Set:

SHAP Decision Plot for SHAP Values in Test Set:

SHAP Decision Plot for Single-Prediction in Test Set:


Checking feature importance for XGB0...

XGB1 In CV1...

Checking if correct model is loaded...
 XGBClassifier(alpha=0.0002926043528872872З, base_score=0.5, booster='gbtree',
              callbacks=None, colsample_bylevel=1, colsample_bynode=1,
              colsample_bytree=0.5441411005619007, early_stopping_rounds=None,
              enable_categorical=False, eta=0.05120369776687421,
              eval_metric=None, gamma=0.4526660690706259, gpu_id=-1,
              grow_policy='depthwise', importance_type=None,
              interaction_constraints='', learning_rate=0.0512036979,
              max_bin=256, max_cat_to_onehot=4, max_delta_step=0, max_depth=18,
              max_leaves=0, min_child_weight=0.12415100550271539,
              min_samples_leaf=9, min_samples_split=27, missing=nan,
              monotone_constraints='()', n_estimators=464, n_jobs=1, nthread=1, ...)
['Age at diagnosis', 'Alanine transaminase (U/L)', 'Albumin (mg/dL)', 'Alcohol', 'Alkaline phosphatase (U/L)', 'Alpha-
Fetoprotein (ng/mL)', 'Ascites degree*', 'Aspartate transaminase (U/L)', 'Cirrhosis', 'Creatinine (mg/dL)', 'Diabete
s', 'Direct Bilirubin (mg/dL)', 'Endemic Countries', 'Esophageal Varices', 'Ferritin (ng/mL)', 'Gamma glutamyl transfe
rase (U/L)', 'Grams of Alcohol per day', 'Haemoglobin (g/dL)', 'Hepatitis B Core Antibody', 'Hepatitis B Surface Antig
en', 'Hepatitis B e Antigen', 'Hepatitis C Virus Antibody', 'International Normalised Ratio*', 'Iron', 'Leukocytes(G/
L)', 'Liver Metastasis', 'Major dimension of nodule (cm)', 'Mean Corpuscular Volume', 'Nonalcoholic Steatohepatitis',
'Number of Nodules', 'Obesity', 'Oxygen Saturation (%)', 'Packs of cigarets per year', 'Performance Status*', 'Platele
ts', 'Portal Hypertension', 'Portal Vein Thrombosis', 'Smoking', 'Symptoms ', 'Total Bilirubin(mg/dL)', 'Total Protein
s (g/dL)']
['Age at diagnosis', 'Alanine transaminase (U/L)', 'Albumin (mg/dL)', 'Alcohol', 'Alkaline phosphatase (U/L)', 'Alpha-
Fetoprotein (ng/mL)', 'Ascites degree*', 'Aspartate transaminase (U/L)', 'Cirrhosis', 'Creatinine (mg/dL)', 'Diabete
s', 'Direct Bilirubin (mg/dL)', 'Endemic Countries', 'Esophageal Varices', 'Ferritin (ng/mL)', 'Gamma glutamyl transfe
rase (U/L)', 'Grams of Alcohol per day', 'Haemoglobin (g/dL)', 'Hepatitis B Core Antibody', 'Hepatitis B Surface Antig
en', 'Hepatitis B e Antigen', 'Hepatitis C Virus Antibody', 'International Normalised Ratio*', 'Iron', 'Leukocytes(G/
L)', 'Liver Metastasis', 'Major dimension of nodule (cm)', 'Mean Corpuscular Volume', 'Nonalcoholic Steatohepatitis',
'Number of Nodules', 'Obesity', 'Oxygen Saturation (%)', 'Packs of cigarets per year', 'Performance Status*', 'Platele
ts', 'Portal Hypertension', 'Portal Vein Thrombosis', 'Smoking', 'Symptoms ', 'Total Bilirubin(mg/dL)', 'Total Protein
s (g/dL)']

Checking explainer for XGB1...
<shap.explainers._tree.Tree object at 0x7f9bb69123d0>

Checking shap values for XGB1...

[[-0.45429307 -0.06374221 -0.94646686 ...  0.2016137  -0.11681356
  -0.14071724]
 [ 0.5519699   0.13245122 -0.10298917 ... -0.83697766 -0.3408652
  -0.07272914]
 [ 0.21548487 -0.07396804  0.14945313 ... -0.72553927  0.11201834
   0.11979318]
 ...
 [-0.237244   -0.04841679 -0.3906867  ... -0.8205372  -0.11913119
  -0.15298116]
 [-0.33601356 -0.06487641  0.14479543 ...  0.26082134  0.02685894
  -0.13012125]
 [ 0.47213364 -0.15212956 -0.5969934  ...  0.20478216 -0.17158583
  -0.18285887]]

Checking shap plots for XGB1...

Expected value for XGB: 1.5605792999267578
Summary Plot for SHAP Values in Test Set:

SHAP Bar Plot for SHAP Values Test Set:

SHAP Decision Plot for SHAP Values in Test Set:
```

```
SHAP Decision Plot for Single-Prediction in Test Set:


Checking feature importance for XGB1...

XGB2 In CV2...

Checking if correct model is loaded...
 XGBClassifier(alpha=5.77534955247629e-07, base_score=0.5, booster='gbtree',
              callbacks=None, colsample_bylevel=1, colsample_bynode=1,
              colsample_bytree=0.41771820514444086, early_stopping_rounds=None,
              enable_categorical=False, eta=8.67291826605322e-06,
              eval_metric=None, gamma=0.07212410933578818, gpu_id=-1,
              grow_policy='lossguide', importance_type=None,
              interaction_constraints='', learning_rate=8.67291874e-06,
              max_bin=256, max_cat_to_onehot=4, max_delta_step=0, max_depth=22,
              max_leaves=0, min_child_weight=6.66045104839759,
              min_samples_leaf=11, min_samples_split=39, missing=nan,
              monotone_constraints='()', n_estimators=884, n_jobs=1, nthread=1, ...)
['Age at diagnosis', 'Albumin (mg/dL)', 'Alcohol', 'Alkaline phosphatase (U/L)', 'Alpha-Fetoprotein (ng/mL)', 'Arteria
l Hypertension', 'Ascites degree*', 'Aspartate transaminase (U/L)', 'Chronic Renal Insufficiency', 'Creatinine (mg/d
L)', 'Diabetes', 'Direct Bilirubin (mg/dL)', 'Encephalopathy degree*', 'Endemic Countries', 'Ferritin (ng/mL)', 'Gamma
glutamyl transferase (U/L)', 'Haemoglobin (g/dL)', 'Hepatitis B Core Antibody', 'Hepatitis B e Antigen', 'Hepatitis C
Virus Antibody', 'Human Immunodeficiency Virus', 'International Normalised Ratio*', 'Iron', 'Leukocytes(G/L)', 'Liver
Metastasis', 'Major dimension of nodule (cm)', 'Mean Corpuscular Volume', 'Number of Nodules', 'Obesity', 'Oxygen Satu
ration (%)', 'Packs of cigarets per year', 'Performance Status*', 'Platelets', 'Portal Hypertension', 'Portal Vein Thr
ombosis', 'Symptoms ']
['Age at diagnosis', 'Albumin (mg/dL)', 'Alcohol', 'Alkaline phosphatase (U/L)', 'Alpha-Fetoprotein (ng/mL)', 'Arteria
l Hypertension', 'Ascites degree*', 'Aspartate transaminase (U/L)', 'Chronic Renal Insufficiency', 'Creatinine (mg/d
L)', 'Diabetes', 'Direct Bilirubin (mg/dL)', 'Encephalopathy degree*', 'Endemic Countries', 'Ferritin (ng/mL)', 'Gamma
glutamyl transferase (U/L)', 'Haemoglobin (g/dL)', 'Hepatitis B Core Antibody', 'Hepatitis B e Antigen', 'Hepatitis C
Virus Antibody', 'Human Immunodeficiency Virus', 'International Normalised Ratio*', 'Iron', 'Leukocytes(G/L)', 'Liver
Metastasis', 'Major dimension of nodule (cm)', 'Mean Corpuscular Volume', 'Number of Nodules', 'Obesity', 'Oxygen Satu
ration (%)', 'Packs of cigarets per year', 'Performance Status*', 'Platelets', 'Portal Hypertension', 'Portal Vein Thr
ombosis', 'Symptoms ']

Checking explainer for XGB2...
<shap.explainers._tree.Tree object at 0x7f9b713e2310>

Checking shap values for XGB2...

[[-1.8894803e-04  2.3255567e-04  0.0000000e+00 ...  0.0000000e+00
  -8.9294117e-06 -7.9064384e-06]
 [-1.9001741e-04  1.9959988e-04  0.0000000e+00 ...  0.0000000e+00
  -1.0370755e-05  3.5941350e-06]
 [-1.6012945e-04  1.6149672e-04  0.0000000e+00 ...  0.0000000e+00
  -1.5008896e-05  3.5941350e-06]
 ...
 [-1.7405280e-04  2.5253580e-04  0.0000000e+00 ...  0.0000000e+00
  -1.0370755e-05  3.5941350e-06]
 [-1.8333328e-04 -4.4896262e-04  0.0000000e+00 ...  0.0000000e+00
  -1.5008896e-05  3.5941350e-06]
 [-1.6480772e-04  2.5592663e-04  0.0000000e+00 ...  0.0000000e+00
  -1.3567553e-05  3.5941350e-06]]

Checking shap plots for XGB2...

Expected value for XGB: -8.377160702366382e-05
Summary Plot for SHAP Values in Test Set:

SHAP Bar Plot for SHAP Values Test Set:

SHAP Decision Plot for SHAP Values in Test Set:

SHAP Decision Plot for Single-Prediction in Test Set:


Checking feature importance for XGB2...

---------------------------------------
hcc-data_example_no_covariates
---------------------------------------
{'Symptoms ': 0, 'Alcohol': 1, 'Hepatitis B Surface Antigen': 2, 'Hepatitis B e Antigen': 3, 'Hepatitis B Core Antibod
y': 4, 'Hepatitis C Virus Antibody': 5, 'Cirrhosis': 6, 'Endemic Countries': 7, 'Smoking': 8, 'Diabetes': 9, 'Obesit
y': 10, 'Hemochromatosis': 11, 'Arterial Hypertension': 12, 'Chronic Renal Insufficiency': 13, 'Human Immunodeficiency
Virus': 14, 'Nonalcoholic Steatohepatitis': 15, 'Esophageal Varices': 16, 'Splenomegaly': 17, 'Portal Hypertension': 1
8, 'Portal Vein Thrombosis': 19, 'Liver Metastasis': 20, 'Radiological Hallmark': 21, 'Grams of Alcohol per day': 22,
'Packs of cigarets per year': 23, 'Performance Status*': 24, 'Encephalopathy degree*': 25, 'Ascites degree*': 26, 'Int
ernational Normalised Ratio*': 27, 'Alpha-Fetoprotein (ng/mL)': 28, 'Haemoglobin (g/dL)': 29, 'Mean Corpuscular Volum
e': 30, 'Leukocytes(G/L)': 31, 'Platelets': 32, 'Albumin (mg/dL)': 33, 'Total Bilirubin(mg/dL)': 34, 'Alanine transami
nase (U/L)': 35, 'Aspartate transaminase (U/L)': 36, 'Gamma glutamyl transferase (U/L)': 37, 'Alkaline phosphatase (U/
L)': 38, 'Total Proteins (g/dL)': 39, 'Creatinine (mg/dL)': 40, 'Number of Nodules': 41, 'Major dimension of nodule (c
m)': 42, 'Direct Bilirubin (mg/dL)': 43, 'Iron': 44, 'Oxygen Saturation (%)': 45, 'Ferritin (ng/mL)': 46}
NB
NB0 In CV0...

Checking if correct model is loaded...
 GaussianNB()
['Alanine transaminase (U/L)', 'Albumin (mg/dL)', 'Alcohol', 'Alkaline phosphatase (U/L)', 'Alpha-Fetoprotein (ng/m
L)', 'Ascites degree*', 'Aspartate transaminase (U/L)', 'Chronic Renal Insufficiency', 'Cirrhosis', 'Creatinine (mg/d
L)', 'Diabetes', 'Direct Bilirubin (mg/dL)', 'Encephalopathy degree*', 'Endemic Countries', 'Esophageal Varices', 'Fer
ritin (ng/mL)', 'Gamma glutamyl transferase (U/L)', 'Haemoglobin (g/dL)', 'Hepatitis B Core Antibody', 'Hepatitis B Su
```

```
rface Antigen', 'Hepatitis C Virus Antibody', 'International Normalised Ratio*', 'Iron', 'Leukocytes(G/L)', 'Liver Met
astasis', 'Major dimension of nodule (cm)', 'Mean Corpuscular Volume', 'Nonalcoholic Steatohepatitis', 'Number of Nodu
les', 'Oxygen Saturation (%)', 'Packs of cigarets per year', 'Performance Status*', 'Portal Vein Thrombosis', 'Splenom
egaly', 'Symptoms ', 'Total Bilirubin(mg/dL)', 'Total Proteins (g/dL)']
['Alanine transaminase (U/L)', 'Albumin (mg/dL)', 'Alcohol', 'Alkaline phosphatase (U/L)', 'Alpha-Fetoprotein (ng/m
L)', 'Ascites degree*', 'Aspartate transaminase (U/L)', 'Chronic Renal Insufficiency', 'Cirrhosis', 'Creatinine (mg/d
L)', 'Diabetes', 'Direct Bilirubin (mg/dL)', 'Encephalopathy degree*', 'Endemic Countries', 'Esophageal Varices', 'Fer
ritin (ng/mL)', 'Gamma glutamyl transferase (U/L)', 'Haemoglobin (g/dL)', 'Hepatitis B Core Antibody', 'Hepatitis B Su
rface Antigen', 'Hepatitis C Virus Antibody', 'International Normalised Ratio*', 'Iron', 'Leukocytes(G/L)', 'Liver Met
astasis', 'Major dimension of nodule (cm)', 'Mean Corpuscular Volume', 'Nonalcoholic Steatohepatitis', 'Number of Nodu
les', 'Oxygen Saturation (%)', 'Packs of cigarets per year', 'Performance Status*', 'Portal Vein Thrombosis', 'Splenom
egaly', 'Symptoms ', 'Total Bilirubin(mg/dL)', 'Total Proteins (g/dL)']

Checking explainer for NB0...
shap.explainers.Permutation()

Checking shap values for NB0...

.values =
array([[ 0.00416667, -0.02      ,  0.        , ...,  0.005     ,
        -0.02333333, -0.03      ],
       [ 0.00333333,  0.00833333,  0.00083333, ...,  0.015     ,
        -0.03416667, -0.0225    ],
       [ 0.03416667,  0.01833333,  0.        , ...,  0.05416667,
        -0.01333333, -0.0275    ],
       ...,
       [ 0.00166667, -0.005     ,  0.        , ...,  0.00666667,
        -0.02833333, -0.0325    ],
       [ 0.00083333, -0.00333333,  0.        , ..., -0.0175    ,
        -0.03      , -0.03      ],
       [ 0.0125    ,  0.02666667,  0.        , ..., -0.045     ,
         0.465     , -0.01833333]])

.base_values =
array([0.32, 0.32, 0.32, 0.32, 0.32, 0.32, 0.32, 0.32, 0.32, 0.32, 0.32,
       0.32, 0.32, 0.32, 0.32, 0.32, 0.32, 0.32, 0.32, 0.32, 0.32, 0.32,
       0.32, 0.32, 0.32, 0.32, 0.32, 0.32, 0.32, 0.32, 0.32, 0.32, 0.32,
       0.32, 0.32, 0.32, 0.32, 0.32, 0.32, 0.32, 0.32, 0.32, 0.32, 0.32,
       0.32, 0.32, 0.32, 0.32, 0.32, 0.32, 0.32, 0.32, 0.32, 0.32, 0.32])

.data =
array([[ 0.0376161,  1.0547612, -1.7113069, ...,  0.6264224, -0.4385377,
        -0.1829094],
       [-0.331668 , -0.2723382, -1.7113069, ...,  0.6264224, -0.3902633,
        -0.3086845],
       [-0.096669 ,  0.3174837, -1.7113069, ...,  0.6264224, -0.486812 ,
        -0.1829094],
       ...,
       [-0.667381 ,  0.6123947,  0.5843487, ...,  0.6264224, -0.3741719,
        -0.1290058],
       [-0.9191656,  0.3174837,  0.5843487, ..., -1.5963668, -0.2615318,
        -0.1379898],
       [ 0.13833  , -0.346066 ,  0.5843487, ..., -1.5963668,  1.154515 ,
        -0.1379898]])

Checking shap plots for NB0...

Summary Plot for SHAP Values in Class 0 & 1 in Test Set:

SHAP Beeswarm Plot for Top 5 SHAP Values in Class 0 & 1 in Test Set:


Checking feature importance for NB0...

NB1 In CV1...

Checking if correct model is loaded...
 GaussianNB()
['Alanine transaminase (U/L)', 'Albumin (mg/dL)', 'Alkaline phosphatase (U/L)', 'Alpha-Fetoprotein (ng/mL)', 'Arterial
Hypertension', 'Ascites degree*', 'Aspartate transaminase (U/L)', 'Cirrhosis', 'Creatinine (mg/dL)', 'Diabetes', 'Dire
ct Bilirubin (mg/dL)', 'Encephalopathy degree*', 'Endemic Countries', 'Ferritin (ng/mL)', 'Gamma glutamyl transferase
(U/L)', 'Haemoglobin (g/dL)', 'Hemochromatosis', 'Hepatitis B e Antigen', 'Hepatitis C Virus Antibody', 'International
Normalised Ratio*', 'Iron', 'Leukocytes(G/L)', 'Liver Metastasis', 'Major dimension of nodule (cm)', 'Mean Corpuscular
Volume', 'Nonalcoholic Steatohepatitis', 'Number of Nodules', 'Obesity', 'Oxygen Saturation (%)', 'Packs of cigarets p
er year', 'Performance Status*', 'Platelets', 'Portal Hypertension', 'Portal Vein Thrombosis', 'Symptoms ', 'Total Bil
irubin(mg/dL)']
['Alanine transaminase (U/L)', 'Albumin (mg/dL)', 'Alkaline phosphatase (U/L)', 'Alpha-Fetoprotein (ng/mL)', 'Arterial
Hypertension', 'Ascites degree*', 'Aspartate transaminase (U/L)', 'Cirrhosis', 'Creatinine (mg/dL)', 'Diabetes', 'Dire
ct Bilirubin (mg/dL)', 'Encephalopathy degree*', 'Endemic Countries', 'Ferritin (ng/mL)', 'Gamma glutamyl transferase
(U/L)', 'Haemoglobin (g/dL)', 'Hemochromatosis', 'Hepatitis B e Antigen', 'Hepatitis C Virus Antibody', 'International
Normalised Ratio*', 'Iron', 'Leukocytes(G/L)', 'Liver Metastasis', 'Major dimension of nodule (cm)', 'Mean Corpuscular
Volume', 'Nonalcoholic Steatohepatitis', 'Number of Nodules', 'Obesity', 'Oxygen Saturation (%)', 'Packs of cigarets p
er year', 'Performance Status*', 'Platelets', 'Portal Hypertension', 'Portal Vein Thrombosis', 'Symptoms ', 'Total Bil
irubin(mg/dL)']

Checking explainer for NB1...
shap.explainers.Permutation()

Checking shap values for NB1...

.values =
array([[ 0.01166667, -0.1075    ,  0.48416667, ..., -0.01666667,
         0.105     , -0.01916667],
       [-0.0025    , -0.005     , -0.03916667, ..., -0.06583333,
        -0.22916667, -0.00833333],
       [ 0.00333333,  0.01916667,  0.04916667, ..., -0.00833333,
        -0.05083333, -0.00916667],
       ...,
       [-0.07166667,  0.00583333,  0.1675    , ..., -0.05166667,
        -0.23166667, -0.0475    ],
       [-0.0075    ,  0.00166667,  0.01333333, ...,  0.01      ,
         0.01833333, -0.0025    ],
       [ 0.01833333, -0.06583333, -0.00666667, ..., -0.01666667,
         0.14333333, -0.01583333]])

.base_values =
array([0.48, 0.48, 0.48, 0.48, 0.48, 0.48, 0.48, 0.48, 0.48, 0.48, 0.48,
       0.48, 0.48, 0.48, 0.48, 0.48, 0.48, 0.48, 0.48, 0.48, 0.48, 0.48,
       0.48, 0.48, 0.48, 0.48, 0.48, 0.48, 0.48, 0.48, 0.48, 0.48, 0.48,
       0.48, 0.48, 0.48, 0.48, 0.48, 0.48, 0.48, 0.48, 0.48, 0.48, 0.48,
       0.48, 0.48, 0.48, 0.48, 0.48, 0.48, 0.48, 0.48, 0.48, 0.48, 0.48])

.data =
array([[-0.3307946,  1.9018224,  2.7240451, ..., -0.5141671,  0.6831301,
        -0.4175834],
       [-0.9380037,  0.0449382, -0.5189802, ..., -0.5141671, -1.4638501,
        -0.449261 ],
       [-0.2159173, -0.5585492,  0.9323539, ..., -0.5141671, -1.4638501,
        -0.1800009],
       ...,
       [ 0.8343903,  0.1996785,  1.465366 , ..., -0.5141671, -1.4638501,
        -0.3700669],
       [ 1.1462003, -0.1098022,  1.3626167, ...,  1.944893 ,  0.6831301,
        -0.2433562],
       [-0.6754268,  1.1281206,  0.1103594, ..., -0.5141671,  0.6831301,
        -0.3859057]])

Checking shap plots for NB1...

Summary Plot for SHAP Values in Class 0 & 1 in Test Set:

SHAP Beeswarm Plot for Top 5 SHAP Values in Class 0 & 1 in Test Set:


Checking feature importance for NB1...

NB2 In CV2...

Checking if correct model is loaded...
 GaussianNB()
['Albumin (mg/dL)', 'Alcohol', 'Alkaline phosphatase (U/L)', 'Alpha-Fetoprotein (ng/mL)', 'Arterial Hypertension', 'As
cites degree*', 'Aspartate transaminase (U/L)', 'Chronic Renal Insufficiency', 'Cirrhosis', 'Creatinine (mg/dL)', 'Dia
betes', 'Direct Bilirubin (mg/dL)', 'Endemic Countries', 'Ferritin (ng/mL)', 'Gamma glutamyl transferase (U/L)', 'Haem
oglobin (g/dL)', 'Hemochromatosis', 'Hepatitis B Surface Antigen', 'Hepatitis B e Antigen', 'Hepatitis C Virus Antibod
y', 'Human Immunodeficiency Virus', 'International Normalised Ratio*', 'Iron', 'Leukocytes(G/L)', 'Liver Metastasis',
'Major dimension of nodule (cm)', 'Mean Corpuscular Volume', 'Nonalcoholic Steatohepatitis', 'Number of Nodules', 'Oxy
gen Saturation (%)', 'Packs of cigarets per year', 'Performance Status*', 'Platelets', 'Portal Hypertension', 'Portal
Vein Thrombosis', 'Smoking']
['Albumin (mg/dL)', 'Alcohol', 'Alkaline phosphatase (U/L)', 'Alpha-Fetoprotein (ng/mL)', 'Arterial Hypertension', 'As
cites degree*', 'Aspartate transaminase (U/L)', 'Chronic Renal Insufficiency', 'Cirrhosis', 'Creatinine (mg/dL)', 'Dia
betes', 'Direct Bilirubin (mg/dL)', 'Endemic Countries', 'Ferritin (ng/mL)', 'Gamma glutamyl transferase (U/L)', 'Haem
oglobin (g/dL)', 'Hemochromatosis', 'Hepatitis B Surface Antigen', 'Hepatitis B e Antigen', 'Hepatitis C Virus Antibod
y', 'Human Immunodeficiency Virus', 'International Normalised Ratio*', 'Iron', 'Leukocytes(G/L)', 'Liver Metastasis',
'Major dimension of nodule (cm)', 'Mean Corpuscular Volume', 'Nonalcoholic Steatohepatitis', 'Number of Nodules', 'Oxy
gen Saturation (%)', 'Packs of cigarets per year', 'Performance Status*', 'Platelets', 'Portal Hypertension', 'Portal
Vein Thrombosis', 'Smoking']

Checking explainer for NB2...
shap.explainers.Permutation()

Checking shap values for NB2...

.values =
array([[ 0.        ,  0.        ,  0.0025    , ...,  0.        ,
         0.        ,  0.        ],
       [ 0.        ,  0.        ,  0.015     , ...,  0.        ,
         0.        ,  0.        ],
       [ 0.00333333, -0.005     ,  0.015     , ..., -0.00333333,
        -0.00333333,  0.0025    ],
       ...,
       [ 0.01416667,  0.0025    ,  0.0325    , ..., -0.0025    ,
        -0.00166667, -0.0075    ],
       [-0.03583333,  0.00583333, -0.00583333, ..., -0.00416667,
        -0.03916667, -0.005     ],
       [ 0.        ,  0.        ,  0.        , ...,  0.        ,
         0.        ,  0.        ]])

.base_values =
array([0.05, 0.05, 0.05, 0.05, 0.05, 0.05, 0.05, 0.05, 0.05, 0.05, 0.05,
       0.05, 0.05, 0.05, 0.05, 0.05, 0.05, 0.05, 0.05, 0.05, 0.05, 0.05,
       0.05, 0.05, 0.05, 0.05, 0.05, 0.05, 0.05, 0.05, 0.05, 0.05, 0.05,
       0.05, 0.05, 0.05, 0.05, 0.05, 0.05, 0.05, 0.05, 0.05, 0.05, 0.05,
       0.05, 0.05, 0.05, 0.05, 0.05, 0.05, 0.05, 0.05, 0.05, 0.05, 0.05])

.data =
array([[-0.4184767,  0.5703518, -0.3573003, ...,  0.6264224, -0.5282705,
        -0.7859052],
       [-0.4184767,  0.5703518,  4.7719674, ..., -1.5963668, -0.5282705,
         1.272418 ],
       [ 0.0205491, -1.7533038,  0.3559594, ...,  0.6264224, -0.5282705,
        -0.7859052],
       ...,
       [-1.8818959,  0.5703518,  1.5964111, ...,  0.6264224, -0.5282705,
         1.272418 ],
       [ 1.0449426,  0.5703518, -0.1898393, ...,  0.6264224, -0.5282705,
         1.272418 ],
       [-2.0282378,  0.5703518, -0.5867838, ...,  0.6264224, -0.5282705,
        -0.7859052]])

Checking shap plots for NB2...

Summary Plot for SHAP Values in Class 0 & 1 in Test Set:

SHAP Beeswarm Plot for Top 5 SHAP Values in Class 0 & 1 in Test Set:


Checking feature importance for NB2...

LR
LR0 In CV0...

Checking if correct model is loaded...
 LogisticRegression(C=0.007632452013609O606, dual=True,
                   max_iter=383.683139958808, random_state=42,
                   solver='liblinear')
['Alanine transaminase (U/L)', 'Albumin (mg/dL)', 'Alcohol', 'Alkaline phosphatase (U/L)', 'Alpha-Fetoprotein (ng/m
L)', 'Ascites degree*', 'Aspartate transaminase (U/L)', 'Chronic Renal Insufficiency', 'Cirrhosis', 'Creatinine (mg/d
L)', 'Diabetes', 'Direct Bilirubin (mg/dL)', 'Encephalopathy degree*', 'Endemic Countries', 'Esophageal Varices', 'Fer
ritin (ng/mL)', 'Gamma glutamyl transferase (U/L)', 'Haemoglobin (g/dL)', 'Hepatitis B Core Antibody', 'Hepatitis B Su
rface Antigen', 'Hepatitis C Virus Antibody', 'International Normalised Ratio*', 'Iron', 'Leukocytes(G/L)', 'Liver Met
astasis', 'Major dimension of nodule (cm)', 'Mean Corpuscular Volume', 'Nonalcoholic Steatohepatitis', 'Number of Nodu
les', 'Oxygen Saturation (%)', 'Packs of cigarets per year', 'Performance Status*', 'Portal Vein Thrombosis', 'Splenom
egaly', 'Symptoms ', 'Total Bilirubin(mg/dL)', 'Total Proteins (g/dL)']
['Alanine transaminase (U/L)', 'Albumin (mg/dL)', 'Alcohol', 'Alkaline phosphatase (U/L)', 'Alpha-Fetoprotein (ng/m
L)', 'Ascites degree*', 'Aspartate transaminase (U/L)', 'Chronic Renal Insufficiency', 'Cirrhosis', 'Creatinine (mg/d
L)', 'Diabetes', 'Direct Bilirubin (mg/dL)', 'Encephalopathy degree*', 'Endemic Countries', 'Esophageal Varices', 'Fer
ritin (ng/mL)', 'Gamma glutamyl transferase (U/L)', 'Haemoglobin (g/dL)', 'Hepatitis B Core Antibody', 'Hepatitis B Su
rface Antigen', 'Hepatitis C Virus Antibody', 'International Normalised Ratio*', 'Iron', 'Leukocytes(G/L)', 'Liver Met
astasis', 'Major dimension of nodule (cm)', 'Mean Corpuscular Volume', 'Nonalcoholic Steatohepatitis', 'Number of Nodu
les', 'Oxygen Saturation (%)', 'Packs of cigarets per year', 'Performance Status*', 'Portal Vein Thrombosis', 'Splenom
egaly', 'Symptoms ', 'Total Bilirubin(mg/dL)', 'Total Proteins (g/dL)']

Checking explainer for LR0...
<shap.explainers._linear.Linear object at 0x7f9b5248c790>

Checking shap values for LR0...

[[-0.00015674 -0.05535253  0.01520368 ...  0.02453187 -0.02176526
  -0.01030959]
 [ 0.00068073  0.02080485  0.01520368 ...  0.02453187 -0.01946545
  -0.01559663]
 [ 0.0001478  -0.01304287  0.01520368 ...  0.02453187 -0.02406506
  -0.01030959]
 ...
 [ 0.00144207 -0.02996674 -0.00683064 ...  0.02453187 -0.01869885
  -0.00804372]
 [ 0.00201307 -0.01304287 -0.00683064 ... -0.0663269  -0.01333263
  -0.00842137]
 [-0.00038514  0.02503582 -0.00683064 ... -0.0663269   0.05412839
  -0.00842137]]

Checking shap plots for LR0...

Expected value for LR: -0.032810633342135964
Summary Plot for SHAP Values in Test Set:

SHAP Bar Plot for SHAP Values Test Set:

SHAP Decision Plot for SHAP Values in Test Set:

SHAP Decision Plot for Single-Prediction in Test Set:


Checking feature importance for LR0...

LR1 In CV1...

Checking if correct model is loaded...
 LogisticRegression(C=0.0003630322466779861, class_weight='balanced',
                   max_iter=159.30522616241012, random_state=42)
['Alanine transaminase (U/L)', 'Albumin (mg/dL)', 'Alkaline phosphatase (U/L)', 'Alpha-Fetoprotein (ng/mL)', 'Arterial
Hypertension', 'Ascites degree*', 'Aspartate transaminase (U/L)', 'Cirrhosis', 'Creatinine (mg/dL)', 'Diabetes', 'Dire
ct Bilirubin (mg/dL)', 'Encephalopathy degree*', 'Endemic Countries', 'Ferritin (ng/mL)', 'Gamma glutamyl transferase
 (U/L)', 'Haemoglobin (g/dL)', 'Hemochromatosis', 'Hepatitis B e Antigen', 'Hepatitis C Virus Antibody', 'Internationa
l Normalised Ratio*', 'Iron', 'Leukocytes(G/L)', 'Liver Metastasis', 'Major dimension of nodule (cm)', 'Mean Corpuscul
ar Volume', 'Nonalcoholic Steatohepatitis', 'Number of Nodules', 'Obesity', 'Oxygen Saturation (%)', 'Packs of cigaret
s per year', 'Performance Status*', 'Platelets', 'Portal Hypertension', 'Portal Vein Thrombosis', 'Symptoms ', 'Total
 Bilirubin(mg/dL)']
['Alanine transaminase (U/L)', 'Albumin (mg/dL)', 'Alkaline phosphatase (U/L)', 'Alpha-Fetoprotein (ng/mL)', 'Arterial
```

```
Hypertension', 'Ascites degree*', 'Aspartate transaminase (U/L)', 'Cirrhosis', 'Creatinine (mg/dL)', 'Diabetes', 'Dire
ct Bilirubin (mg/dL)', 'Encephalopathy degree*', 'Endemic Countries', 'Ferritin (ng/mL)', 'Gamma glutamyl transferase
 (U/L)', 'Haemoglobin (g/dL)', 'Hemochromatosis', 'Hepatitis B e Antigen', 'Hepatitis C Virus Antibody', 'Internationa
l Normalised Ratio*', 'Iron', 'Leukocytes(G/L)', 'Liver Metastasis', 'Major dimension of nodule (cm)', 'Mean Corpuscul
ar Volume', 'Nonalcoholic Steatohepatitis', 'Number of Nodules', 'Obesity', 'Oxygen Saturation (%)', 'Packs of cigaret
s per year', 'Performance Status*', 'Platelets', 'Portal Hypertension', 'Portal Vein Thrombosis', 'Symptoms ', 'Total
 Bilirubin(mg/dL)']

Checking explainer for LR1...
<shap.explainers._linear.Linear object at 0x7f9b7264c670>

Checking shap values for LR1...

[[ 0.00034788 -0.01330814  0.01706337 ... -0.00222993  0.00364155
  -0.00301374]
 [ 0.00099548 -0.00012806 -0.00335686 ... -0.00222993 -0.00984567
  -0.00321245]
 [ 0.00022536  0.00415547  0.0057817  ... -0.00222993 -0.00984567
  -0.00152343]
 ...
 [-0.00089483 -0.0012264   0.00913789 ... -0.00222993 -0.00984567
  -0.00271568]
 [-0.00122738  0.00097028  0.00849092 ...  0.00602908  0.00364155
  -0.00192084]
 [ 0.00071544 -0.00781644  0.00060588 ... -0.00222993  0.00364155
  -0.00281503]]

Checking shap plots for LR1...

Expected value for LR: -0.004693505131985552
Summary Plot for SHAP Values in Test Set:

SHAP Bar Plot for SHAP Values Test Set:

SHAP Decision Plot for SHAP Values in Test Set:

SHAP Decision Plot for Single-Prediction in Test Set:


Checking feature importance for LR1...

LR2 In CV2...

Checking if correct model is loaded...
 LogisticRegression(C=1.1355889848345609e-05, dual=True,
                   max_iter=287.08753481954665, random_state=42,
                   solver='liblinear')
['Albumin (mg/dL)', 'Alcohol', 'Alkaline phosphatase (U/L)', 'Alpha-Fetoprotein (ng/mL)', 'Arterial Hypertension', 'As
cites degree*', 'Aspartate transaminase (U/L)', 'Chronic Renal Insufficiency', 'Cirrhosis', 'Creatinine (mg/dL)', 'Dia
betes', 'Direct Bilirubin (mg/dL)', 'Endemic Countries', 'Ferritin (ng/mL)', 'Gamma glutamyl transferase (U/L)', 'Haem
oglobin (g/dL)', 'Hemochromatosis', 'Hepatitis B Surface Antigen', 'Hepatitis B e Antigen', 'Hepatitis C Virus Antibod
y', 'Human Immunodeficiency Virus', 'International Normalised Ratio*', 'Iron', 'Leukocytes(G/L)', 'Liver Metastasis',
 'Major dimension of nodule (cm)', 'Mean Corpuscular Volume', 'Nonalcoholic Steatohepatitis', 'Number of Nodules', 'Ox
ygen Saturation (%)', 'Packs of cigarets per year', 'Performance Status*', 'Platelets', 'Portal Hypertension', 'Portal
Vein Thrombosis', 'Smoking']
['Albumin (mg/dL)', 'Alcohol', 'Alkaline phosphatase (U/L)', 'Alpha-Fetoprotein (ng/mL)', 'Arterial Hypertension', 'As
cites degree*', 'Aspartate transaminase (U/L)', 'Chronic Renal Insufficiency', 'Cirrhosis', 'Creatinine (mg/dL)', 'Dia
betes', 'Direct Bilirubin (mg/dL)', 'Endemic Countries', 'Ferritin (ng/mL)', 'Gamma glutamyl transferase (U/L)', 'Haem
oglobin (g/dL)', 'Hemochromatosis', 'Hepatitis B Surface Antigen', 'Hepatitis B e Antigen', 'Hepatitis C Virus Antibod
y', 'Human Immunodeficiency Virus', 'International Normalised Ratio*', 'Iron', 'Leukocytes(G/L)', 'Liver Metastasis',
 'Major dimension of nodule (cm)', 'Mean Corpuscular Volume', 'Nonalcoholic Steatohepatitis', 'Number of Nodules', 'Ox
ygen Saturation (%)', 'Packs of cigarets per year', 'Performance Status*', 'Platelets', 'Portal Hypertension', 'Portal
Vein Thrombosis', 'Smoking']

Checking explainer for LR2...
<shap.explainers._linear.Linear object at 0x7f9b730a2160>

Checking shap values for LR2...

[[ 7.32083460e-05  1.92822706e-05 -4.36797008e-05 ... -3.15759647e-05
  -7.30478160e-05  3.36532768e-05]
 [ 7.32083460e-05  1.92822706e-05  8.74173101e-04 ...  8.11953378e-05
  -7.30478160e-05 -5.73015253e-05]
 [ 7.45330978e-06 -7.71290826e-05  8.39539991e-05 ... -3.15759647e-05
  -7.30478160e-05  3.36532768e-05]
 ...
 [ 2.92391780e-04  1.92822706e-05  3.05925660e-04 ... -3.15759647e-05
  -7.30478160e-05 -5.73015253e-05]
 [-1.45975103e-04  1.92822706e-05 -1.37135230e-05 ... -3.15759647e-05
  -7.30478160e-05 -5.73015253e-05]
 [ 3.14310121e-04  1.92822706e-05 -8.47444465e-05 ... -3.15759647e-05
  -7.30478160e-05  3.36532768e-05]]

Checking shap plots for LR2...

Expected value for LR: -0.0002947812564940089
Summary Plot for SHAP Values in Test Set:

SHAP Bar Plot for SHAP Values Test Set:

SHAP Decision Plot for SHAP Values in Test Set:

SHAP Decision Plot for Single-Prediction in Test Set:
```

```
Checking feature importance for LR2...

DT
DT0 In CV0...

Checking if correct model is loaded...
 DecisionTreeClassifier(max_depth=17, min_samples_leaf=35, min_samples_split=45,
                       random_state=42)
['Alanine transaminase (U/L)', 'Albumin (mg/dL)', 'Alcohol', 'Alkaline phosphatase (U/L)', 'Alpha-Fetoprotein (ng/m
L)', 'Ascites degree*', 'Aspartate transaminase (U/L)', 'Chronic Renal Insufficiency', 'Cirrhosis', 'Creatinine (mg/d
L)', 'Diabetes', 'Direct Bilirubin (mg/dL)', 'Encephalopathy degree*', 'Endemic Countries', 'Esophageal Varices', 'Fer
ritin (ng/mL)', 'Gamma glutamyl transferase (U/L)', 'Haemoglobin (g/dL)', 'Hepatitis B Core Antibody', 'Hepatitis B Su
rface Antigen', 'Hepatitis C Virus Antibody', 'International Normalised Ratio*', 'Iron', 'Leukocytes(G/L)', 'Liver Met
astasis', 'Major dimension of nodule (cm)', 'Mean Corpuscular Volume', 'Nonalcoholic Steatohepatitis', 'Number of Nodu
les', 'Oxygen Saturation (%)', 'Packs of cigarets per year', 'Performance Status*', 'Portal Vein Thrombosis', 'Splenom
egaly', 'Symptoms ', 'Total Bilirubin(mg/dL)', 'Total Proteins (g/dL)']
['Alanine transaminase (U/L)', 'Albumin (mg/dL)', 'Alcohol', 'Alkaline phosphatase (U/L)', 'Alpha-Fetoprotein (ng/m
L)', 'Ascites degree*', 'Aspartate transaminase (U/L)', 'Chronic Renal Insufficiency', 'Cirrhosis', 'Creatinine (mg/d
L)', 'Diabetes', 'Direct Bilirubin (mg/dL)', 'Encephalopathy degree*', 'Endemic Countries', 'Esophageal Varices', 'Fer
ritin (ng/mL)', 'Gamma glutamyl transferase (U/L)', 'Haemoglobin (g/dL)', 'Hepatitis B Core Antibody', 'Hepatitis B Su
rface Antigen', 'Hepatitis C Virus Antibody', 'International Normalised Ratio*', 'Iron', 'Leukocytes(G/L)', 'Liver Met
astasis', 'Major dimension of nodule (cm)', 'Mean Corpuscular Volume', 'Nonalcoholic Steatohepatitis', 'Number of Nodu
les', 'Oxygen Saturation (%)', 'Packs of cigarets per year', 'Performance Status*', 'Portal Vein Thrombosis', 'Splenom
egaly', 'Symptoms ', 'Total Bilirubin(mg/dL)', 'Total Proteins (g/dL)']

Checking explainer for DT0...
<shap.explainers._tree.Tree object at 0x7f9b7390df70>

Checking shap values for DT0...

[array([[0., 0., 0., ..., 0., 0., 0.],
       [0., 0., 0., ..., 0., 0., 0.],
       [0., 0., 0., ..., 0., 0., 0.],
       ...,
       [0., 0., 0., ..., 0., 0., 0.],
       [0., 0., 0., ..., 0., 0., 0.],
       [0., 0., 0., ..., 0., 0., 0.]]), array([[0., 0., 0., ..., 0., 0., 0.],
       [0., 0., 0., ..., 0., 0., 0.],
       [0., 0., 0., ..., 0., 0., 0.],
       ...,
       [0., 0., 0., ..., 0., 0., 0.],
       [0., 0., 0., ..., 0., 0., 0.],
       [0., 0., 0., ..., 0., 0., 0.]])]

Checking shap plots for DT0...

Expected value for DT: [0.57272727 0.42727273]
Bar Summary Plot for SHAP Values in Class 0 & 1 in Test Set:


Decision Plot for SHAP Values from Class 0 in Test Set:


Decision Plot for SHAP Values from Class 1 in Test Set:


Checking feature importance for DT0...

DT1 In CV1...

Checking if correct model is loaded...
 DecisionTreeClassifier(criterion='entropy', max_depth=30, min_samples_leaf=20,
                       min_samples_split=5, random_state=42, splitter='random')
['Alanine transaminase (U/L)', 'Albumin (mg/dL)', 'Alkaline phosphatase (U/L)', 'Alpha-Fetoprotein (ng/mL)', 'Arterial
Hypertension', 'Ascites degree*', 'Aspartate transaminase (U/L)', 'Cirrhosis', 'Creatinine (mg/dL)', 'Diabetes', 'Dire
ct Bilirubin (mg/dL)', 'Encephalopathy degree*', 'Endemic Countries', 'Ferritin (ng/mL)', 'Gamma glutamyl transferase
 (U/L)', 'Haemoglobin (g/dL)', 'Hemochromatosis', 'Hepatitis B e Antigen', 'Hepatitis C Virus Antibody', 'Internationa
l Normalised Ratio*', 'Iron', 'Leukocytes(G/L)', 'Liver Metastasis', 'Major dimension of nodule (cm)', 'Mean Corpuscul
ar Volume', 'Nonalcoholic Steatohepatitis', 'Number of Nodules', 'Obesity', 'Oxygen Saturation (%)', 'Packs of cigaret
s per year', 'Performance Status*', 'Platelets', 'Portal Hypertension', 'Portal Vein Thrombosis', 'Symptoms ', 'Total
 Bilirubin(mg/dL)']
['Alanine transaminase (U/L)', 'Albumin (mg/dL)', 'Alkaline phosphatase (U/L)', 'Alpha-Fetoprotein (ng/mL)', 'Arterial
Hypertension', 'Ascites degree*', 'Aspartate transaminase (U/L)', 'Cirrhosis', 'Creatinine (mg/dL)', 'Diabetes', 'Dire
ct Bilirubin (mg/dL)', 'Encephalopathy degree*', 'Endemic Countries', 'Ferritin (ng/mL)', 'Gamma glutamyl transferase
 (U/L)', 'Haemoglobin (g/dL)', 'Hemochromatosis', 'Hepatitis B e Antigen', 'Hepatitis C Virus Antibody', 'Internationa
l Normalised Ratio*', 'Iron', 'Leukocytes(G/L)', 'Liver Metastasis', 'Major dimension of nodule (cm)', 'Mean Corpuscul
ar Volume', 'Nonalcoholic Steatohepatitis', 'Number of Nodules', 'Obesity', 'Oxygen Saturation (%)', 'Packs of cigaret
s per year', 'Performance Status*', 'Platelets', 'Portal Hypertension', 'Portal Vein Thrombosis', 'Symptoms ', 'Total
 Bilirubin(mg/dL)']

Checking explainer for DT1...
<shap.explainers._tree.Tree object at 0x7f9b5514aaf0>

Checking shap values for DT1...

[array([[ 0.        ,  0.08222611,  0.        , ...,  0.        ,
          0.        ,  0.        ],
       [ 0.        , -0.02219251,  0.        , ...,  0.        ,
          0.        ,  0.        ],
       [ 0.        , -0.02219251,  0.        , ...,  0.        ,
          0.        ,  0.        ],
       ...,
       [ 0.        , -0.02219251,  0.        , ...,  0.        ,
         0.        ,  0.        ],
       [ 0.        , -0.06287879,  0.        , ...,  0.        ,
         0.        ,  0.        ],
       [ 0.        ,  0.02902098,  0.        , ...,  0.        ,
         0.        ,  0.        ]]), array([[ 0.        , -0.08222611,  0.        , ...,  0.        ,
         0.        ,  0.        ],
       [ 0.        ,  0.02219251,  0.        , ...,  0.        ,
         0.        ,  0.        ],
       [ 0.        ,  0.02219251,  0.        , ...,  0.        ,
         0.        ,  0.        ],
       ...,
       [ 0.        ,  0.02219251,  0.        , ...,  0.        ,
         0.        ,  0.        ],
       [ 0.        ,  0.06287879,  0.        , ...,  0.        ,
         0.        ,  0.        ],
       [ 0.        , -0.02902098,  0.        , ...,  0.        ,
         0.        ,  0.        ]])]

Checking shap plots for DT1...

Expected value for DT: [0.63636364 0.36363636]
Bar Summary Plot for SHAP Values in Class 0 & 1 in Test Set:


Decision Plot for SHAP Values from Class 0 in Test Set:


Decision Plot for SHAP Values from Class 1 in Test Set:


Checking feature importance for DT1...

DT2 In CV2...

Checking if correct model is loaded...
 DecisionTreeClassifier(class_weight='balanced', max_depth=29,
                       min_samples_leaf=30, min_samples_split=45,
                       random_state=42)
['Albumin (mg/dL)', 'Alcohol', 'Alkaline phosphatase (U/L)', 'Alpha-Fetoprotein (ng/mL)', 'Arterial Hypertension', 'As
cites degree*', 'Aspartate transaminase (U/L)', 'Chronic Renal Insufficiency', 'Cirrhosis', 'Creatinine (mg/dL)', 'Dia
betes', 'Direct Bilirubin (mg/dL)', 'Endemic Countries', 'Ferritin (ng/mL)', 'Gamma glutamyl transferase (U/L)', 'Haem
oglobin (g/dL)', 'Hemochromatosis', 'Hepatitis B Surface Antigen', 'Hepatitis B e Antigen', 'Hepatitis C Virus Antibod
y', 'Human Immunodeficiency Virus', 'International Normalised Ratio*', 'Iron', 'Leukocytes(G/L)', 'Liver Metastasis',
 'Major dimension of nodule (cm)', 'Mean Corpuscular Volume', 'Nonalcoholic Steatohepatitis', 'Number of Nodules', 'Ox
ygen Saturation (%)', 'Packs of cigarets per year', 'Performance Status*', 'Platelets', 'Portal Hypertension', 'Portal
Vein Thrombosis', 'Smoking']
['Albumin (mg/dL)', 'Alcohol', 'Alkaline phosphatase (U/L)', 'Alpha-Fetoprotein (ng/mL)', 'Arterial Hypertension', 'As
cites degree*', 'Aspartate transaminase (U/L)', 'Chronic Renal Insufficiency', 'Cirrhosis', 'Creatinine (mg/dL)', 'Dia
betes', 'Direct Bilirubin (mg/dL)', 'Endemic Countries', 'Ferritin (ng/mL)', 'Gamma glutamyl transferase (U/L)', 'Haem
oglobin (g/dL)', 'Hemochromatosis', 'Hepatitis B Surface Antigen', 'Hepatitis B e Antigen', 'Hepatitis C Virus Antibod
y', 'Human Immunodeficiency Virus', 'International Normalised Ratio*', 'Iron', 'Leukocytes(G/L)', 'Liver Metastasis',
 'Major dimension of nodule (cm)', 'Mean Corpuscular Volume', 'Nonalcoholic Steatohepatitis', 'Number of Nodules', 'Ox
ygen Saturation (%)', 'Packs of cigarets per year', 'Performance Status*', 'Platelets', 'Portal Hypertension', 'Portal
Vein Thrombosis', 'Smoking']

Checking explainer for DT2...
<shap.explainers._tree.Tree object at 0x7f9ba112c940>

Checking shap values for DT2...

[array([[-0.14385676,  0.        ,  0.20558515, ...,  0.        ,
         0.        ,  0.        ],
       [-0.0462963 ,  0.        , -0.18566247, ...,  0.        ,
         0.        ,  0.        ],
       [-0.0462963 ,  0.        , -0.18566247, ...,  0.        ,
         0.        ,  0.        ],
       ...,
       [-0.0462963 ,  0.        , -0.18566247, ...,  0.        ,
         0.        ,  0.        ],
       [ 0.05769231,  0.        , -0.28965107, ...,  0.        ,
         0.        ,  0.        ],
       [-0.14385676,  0.        ,  0.20558515, ...,  0.        ,
         0.        ,  0.        ]]), array([[ 0.14385676,  0.        , -0.20558515, ...,  0.        ,
         0.        ,  0.        ],
       [ 0.0462963 ,  0.        ,  0.18566247, ...,  0.        ,
         0.        ,  0.        ],
       [ 0.0462963 ,  0.        ,  0.18566247, ...,  0.        ,
         0.        ,  0.        ],
       ...,
       [ 0.0462963 ,  0.        ,  0.18566247, ...,  0.        ,
         0.        ,  0.        ],
       [-0.05769231,  0.        ,  0.28965107, ...,  0.        ,
         0.        ,  0.        ],
       [ 0.14385676,  0.        , -0.20558515, ...,  0.        ,
         0.        ,  0.        ]])]

Checking shap plots for DT2...

Expected value for DT: [0.5 0.5]
Bar Summary Plot for SHAP Values in Class 0 & 1 in Test Set:
```

```
Decision Plot for SHAP Values from Class 0 in Test Set:


Decision Plot for SHAP Values from Class 1 in Test Set:


Checking feature importance for DT2...

RF
RF0 In CV0...

Checking if correct model is loaded...
 RandomForestClassifier(max_depth=9, max_features=None, min_samples_leaf=9,
                       min_samples_split=24, n_estimators=935, random_state=42)
['Alanine transaminase (U/L)', 'Albumin (mg/dL)', 'Alcohol', 'Alkaline phosphatase (U/L)', 'Alpha-Fetoprotein (ng/m
L)', 'Ascites degree*', 'Aspartate transaminase (U/L)', 'Chronic Renal Insufficiency', 'Cirrhosis', 'Creatinine (mg/d
L)', 'Diabetes', 'Direct Bilirubin (mg/dL)', 'Encephalopathy degree*', 'Endemic Countries', 'Esophageal Varices', 'Fer
ritin (ng/mL)', 'Gamma glutamyl transferase (U/L)', 'Haemoglobin (g/dL)', 'Hepatitis B Core Antibody', 'Hepatitis B Su
rface Antigen', 'Hepatitis C Virus Antibody', 'International Normalised Ratio*', 'Iron', 'Leukocytes(G/L)', 'Liver Met
astasis', 'Major dimension of nodule (cm)', 'Mean Corpuscular Volume', 'Nonalcoholic Steatohepatitis', 'Number of Nodu
les', 'Oxygen Saturation (%)', 'Packs of cigarets per year', 'Performance Status*', 'Portal Vein Thrombosis', 'Splenom
egaly', 'Symptoms ', 'Total Bilirubin(mg/dL)', 'Total Proteins (g/dL)']
['Alanine transaminase (U/L)', 'Albumin (mg/dL)', 'Alcohol', 'Alkaline phosphatase (U/L)', 'Alpha-Fetoprotein (ng/m
L)', 'Ascites degree*', 'Aspartate transaminase (U/L)', 'Chronic Renal Insufficiency', 'Cirrhosis', 'Creatinine (mg/d
L)', 'Diabetes', 'Direct Bilirubin (mg/dL)', 'Encephalopathy degree*', 'Endemic Countries', 'Esophageal Varices', 'Fer
ritin (ng/mL)', 'Gamma glutamyl transferase (U/L)', 'Haemoglobin (g/dL)', 'Hepatitis B Core Antibody', 'Hepatitis B Su
rface Antigen', 'Hepatitis C Virus Antibody', 'International Normalised Ratio*', 'Iron', 'Leukocytes(G/L)', 'Liver Met
astasis', 'Major dimension of nodule (cm)', 'Mean Corpuscular Volume', 'Nonalcoholic Steatohepatitis', 'Number of Nodu
les', 'Oxygen Saturation (%)', 'Packs of cigarets per year', 'Performance Status*', 'Portal Vein Thrombosis', 'Splenom
egaly', 'Symptoms ', 'Total Bilirubin(mg/dL)', 'Total Proteins (g/dL)']

Checking explainer for RF0...
<shap.explainers._tree.Tree object at 0x7f9b5528e4c0>

Checking shap values for RF0...

[array([[-8.04892952e-04,  3.49123556e-02, -7.39028100e-04, ...,
         -2.55865722e-05, -4.80373900e-04, -5.51427280e-03],
        [ 8.98581287e-04, -2.36897659e-02, -7.01881775e-05, ...,
         -4.69087156e-05, -1.53604128e-03, -9.93491258e-03],
        [ 1.09297656e-04, -2.10001659e-02,  2.39694450e-04, ...,
         -4.18242045e-05, -3.61466526e-04, -6.98520090e-03],
        ...,
        [ 3.18715672e-04,  2.90540264e-02,  2.67651465e-04, ...,
         -4.18242045e-05,  7.09998438e-05,  7.60365635e-03],
        [-1.87966055e-03, -1.58333798e-02,  2.67651465e-04, ...,
          3.18309142e-04,  1.93293842e-03, -7.31627298e-04],
        [-1.82586614e-03, -2.32531905e-02,  1.25596859e-04, ...,
          3.18309142e-04, -1.88973059e-03, -1.80372814e-03]]), array([[ 8.04892952e-04, -3.49123556e-02,  7.39028100e-0
4, ...,
          2.55865722e-05,  4.80373900e-04,  5.51427280e-03],
        [-8.98581287e-04,  2.36897659e-02,  7.01881775e-05, ...,
          4.69087156e-05,  1.53604128e-03,  9.93491258e-03],
        [-1.09297656e-04,  2.10001659e-02, -2.39694450e-04, ...,
          4.18242045e-05,  3.61466526e-04,  6.98520090e-03],
        ...,
        [-3.18715672e-04, -2.90540264e-02, -2.67651465e-04, ...,
          4.18242045e-05, -7.09998438e-05, -7.60365635e-03],
        [ 1.87966055e-03,  1.58333798e-02, -2.67651465e-04, ...,
         -3.18309142e-04, -1.93293842e-03,  7.31627298e-04],
        [ 1.82586614e-03,  2.32531905e-02, -1.25596859e-04, ...,
         -3.18309142e-04,  1.88973059e-03,  1.80372814e-03]])]

Checking shap plots for RF0...

Expected value for RF: [0.57252309 0.42747691]
Bar Summary Plot for SHAP Values in Class 0 & 1 in Test Set:


Decision Plot for SHAP Values from Class 0 in Test Set:


Decision Plot for SHAP Values from Class 1 in Test Set:


Checking feature importance for RF0...

RF1 In CV1...

Checking if correct model is loaded...
 RandomForestClassifier(class_weight='balanced', max_depth=25,
                       max_features='log2', min_samples_leaf=16,
                       min_samples_split=45, n_estimators=194, oob_score=True,
                       random_state=42)
['Alanine transaminase (U/L)', 'Albumin (mg/dL)', 'Alkaline phosphatase (U/L)', 'Alpha-Fetoprotein (ng/mL)', 'Arterial
Hypertension', 'Ascites degree*', 'Aspartate transaminase (U/L)', 'Cirrhosis', 'Creatinine (mg/dL)', 'Diabetes', 'Dire
ct Bilirubin (mg/dL)', 'Encephalopathy degree*', 'Endemic Countries', 'Ferritin (ng/mL)', 'Gamma glutamyl transferase
 (U/L)', 'Haemoglobin (g/dL)', 'Hemochromatosis', 'Hepatitis B e Antigen', 'Hepatitis C Virus Antibody', 'Internationa
l Normalised Ratio*', 'Iron', 'Leukocytes(G/L)', 'Liver Metastasis', 'Major dimension of nodule (cm)', 'Mean Corpuscul
ar Volume', 'Nonalcoholic Steatohepatitis', 'Number of Nodules', 'Obesity', 'Oxygen Saturation (%)', 'Packs of cigaret
s per year', 'Performance Status*', 'Platelets', 'Portal Hypertension', 'Portal Vein Thrombosis', 'Symptoms ', 'Total
 Bilirubin(mg/dL)']
```

```
['Alanine transaminase (U/L)', 'Albumin (mg/dL)', 'Alkaline phosphatase (U/L)', 'Alpha-Fetoprotein (ng/mL)', 'Arterial
Hypertension', 'Ascites degree*', 'Aspartate transaminase (U/L)', 'Cirrhosis', 'Creatinine (mg/dL)', 'Diabetes', 'Dire
ct Bilirubin (mg/dL)', 'Encephalopathy degree*', 'Endemic Countries', 'Ferritin (ng/mL)', 'Gamma glutamyl transferase
 (U/L)', 'Haemoglobin (g/dL)', 'Hemochromatosis', 'Hepatitis B e Antigen', 'Hepatitis C Virus Antibody', 'Internationa
l Normalised Ratio*', 'Iron', 'Leukocytes(G/L)', 'Liver Metastasis', 'Major dimension of nodule (cm)', 'Mean Corpuscul
ar Volume', 'Nonalcoholic Steatohepatitis', 'Number of Nodules', 'Obesity', 'Oxygen Saturation (%)', 'Packs of cigaret
s per year', 'Performance Status*', 'Platelets', 'Portal Hypertension', 'Portal Vein Thrombosis', 'Symptoms ', 'Total
 Bilirubin(mg/dL)']

Checking explainer for RF1...
<shap.explainers._tree.Tree object at 0x7f9b557c64f0>

Checking shap values for RF1...

[array([[ 6.89282193e-04,  5.16384335e-02, -1.76947858e-02, ...,
         4.15186605e-04, -5.22220942e-03,  1.52801529e-02],
       [-2.86516996e-03, -1.75170521e-03,  1.53660207e-02, ...,
         1.57204645e-04,  1.14384010e-02,  1.36293862e-02],
       [ 9.58343029e-04, -3.08481544e-02, -1.63920514e-02, ...,
         1.57204645e-04,  1.17329978e-02, -6.96607661e-03],
       ...,
       [-4.62392460e-05,  2.51854494e-02, -1.70193937e-02, ...,
         1.57204645e-04,  1.17917953e-02,  1.21731827e-02],
       [ 8.00804127e-04, -3.08295224e-02, -1.58989779e-02, ...,
        -1.04418188e-03, -5.37882055e-03, -6.99630666e-04],
       [-1.58770860e-03,  5.12094387e-02, -9.49257495e-03, ...,
         1.57204645e-04, -5.09113830e-03,  1.43028838e-02]]), array([[-6.89282193e-04, -5.16384335e-02,  1.76947858e-0
2, ...,
        -4.15186605e-04,  5.22220942e-03, -1.52801529e-02],
       [ 2.86516996e-03,  1.75170521e-03, -1.53660207e-02, ...,
        -1.57204645e-04, -1.14384010e-02, -1.36293862e-02],
       [-9.58343029e-04,  3.08481544e-02,  1.63920514e-02, ...,
        -1.57204645e-04, -1.17329978e-02,  6.96607661e-03],
       ...,
       [ 4.62392460e-05, -2.51854494e-02,  1.70193937e-02, ...,
        -1.57204645e-04, -1.17917953e-02, -1.21731827e-02],
       [-8.00804127e-04,  3.08295224e-02,  1.58989779e-02, ...,
         1.04418188e-03,  5.37882055e-03,  6.99630666e-04],
       [ 1.58770860e-03, -5.12094387e-02,  9.49257495e-03, ...,
        -1.57204645e-04,  5.09113830e-03, -1.43028838e-02]])]

Checking shap plots for RF1...

Expected value for RF: [0.49611628 0.50388372]
Bar Summary Plot for SHAP Values in Class 0 & 1 in Test Set:


Decision Plot for SHAP Values from Class 0 in Test Set:


Decision Plot for SHAP Values from Class 1 in Test Set:


Checking feature importance for RF1...

RF2 In CV2...

Checking if correct model is loaded...
 RandomForestClassifier(class_weight='balanced', max_depth=16, max_features=None,
                       min_samples_leaf=20, min_samples_split=40,
                       n_estimators=299, oob_score=True, random_state=42)
['Albumin (mg/dL)', 'Alcohol', 'Alkaline phosphatase (U/L)', 'Alpha-Fetoprotein (ng/mL)', 'Arterial Hypertension', 'As
cites degree*', 'Aspartate transaminase (U/L)', 'Chronic Renal Insufficiency', 'Cirrhosis', 'Creatinine (mg/dL)', 'Dia
betes', 'Direct Bilirubin (mg/dL)', 'Endemic Countries', 'Ferritin (ng/mL)', 'Gamma glutamyl transferase (U/L)', 'Haem
oglobin (g/dL)', 'Hemochromatosis', 'Hepatitis B Surface Antigen', 'Hepatitis B e Antigen', 'Hepatitis C Virus Antibod
y', 'Human Immunodeficiency Virus', 'International Normalised Ratio*', 'Iron', 'Leukocytes(G/L)', 'Liver Metastasis',
 'Major dimension of nodule (cm)', 'Mean Corpuscular Volume', 'Nonalcoholic Steatohepatitis', 'Number of Nodules', 'Ox
ygen Saturation (%)', 'Packs of cigarets per year', 'Performance Status*', 'Platelets', 'Portal Hypertension', 'Portal
Vein Thrombosis', 'Smoking']
['Albumin (mg/dL)', 'Alcohol', 'Alkaline phosphatase (U/L)', 'Alpha-Fetoprotein (ng/mL)', 'Arterial Hypertension', 'As
cites degree*', 'Aspartate transaminase (U/L)', 'Chronic Renal Insufficiency', 'Cirrhosis', 'Creatinine (mg/dL)', 'Dia
betes', 'Direct Bilirubin (mg/dL)', 'Endemic Countries', 'Ferritin (ng/mL)', 'Gamma glutamyl transferase (U/L)', 'Haem
oglobin (g/dL)', 'Hemochromatosis', 'Hepatitis B Surface Antigen', 'Hepatitis B e Antigen', 'Hepatitis C Virus Antibod
y', 'Human Immunodeficiency Virus', 'International Normalised Ratio*', 'Iron', 'Leukocytes(G/L)', 'Liver Metastasis',
 'Major dimension of nodule (cm)', 'Mean Corpuscular Volume', 'Nonalcoholic Steatohepatitis', 'Number of Nodules', 'Ox
ygen Saturation (%)', 'Packs of cigarets per year', 'Performance Status*', 'Platelets', 'Portal Hypertension', 'Portal
Vein Thrombosis', 'Smoking']

Checking explainer for RF2...
<shap.explainers._tree.Tree object at 0x7f9b557e1430>

Checking shap values for RF2...

[array([[-1.64930049e-02,  0.00000000e+00,  9.65987236e-02, ...,
         0.00000000e+00,  0.00000000e+00, -2.92612124e-04],
       [-1.10170786e-02,  0.00000000e+00, -1.61874431e-01, ...,
         0.00000000e+00,  0.00000000e+00,  4.35547992e-04],
       [-1.17523110e-02,  0.00000000e+00, -1.57953243e-01, ...,
         0.00000000e+00,  0.00000000e+00, -2.92612124e-04],
       ...,
       [-1.24810033e-02,  0.00000000e+00, -1.55756997e-01, ...,
         0.00000000e+00,  0.00000000e+00,  4.35547992e-04],
       [ 2.57852085e-02,  0.00000000e+00, -8.08842985e-02, ...,
         0.00000000e+00,  0.00000000e+00,  4.35547992e-04],
       [-1.64930049e-02,  0.00000000e+00,  1.95650937e-01, ...,
         0.00000000e+00,  0.00000000e+00, -1.15877556e-04]]), array([[ 1.64930049e-02,  0.00000000e+00, -9.65987236e-0
2, ...,
         0.00000000e+00,  0.00000000e+00,  2.92612124e-04],
       [ 1.10170786e-02,  0.00000000e+00,  1.61874431e-01, ...,
         0.00000000e+00,  0.00000000e+00, -4.35547992e-04],
       [ 1.17523110e-02,  0.00000000e+00,  1.57953243e-01, ...,
         0.00000000e+00,  0.00000000e+00,  2.92612124e-04],
       ...,
       [ 1.24810033e-02,  0.00000000e+00,  1.55756997e-01, ...,
         0.00000000e+00,  0.00000000e+00, -4.35547992e-04],
       [-2.57852085e-02,  0.00000000e+00,  8.08842985e-02, ...,
         0.00000000e+00,  0.00000000e+00, -4.35547992e-04],
       [ 1.64930049e-02,  0.00000000e+00, -1.95650937e-01, ...,
         0.00000000e+00,  0.00000000e+00,  1.15877556e-04]]])]

Checking shap plots for RF2...

Expected value for RF: [0.49969365 0.50030635]
Bar Summary Plot for SHAP Values in Class 0 & 1 in Test Set:


Decision Plot for SHAP Values from Class 0 in Test Set:


Decision Plot for SHAP Values from Class 1 in Test Set:


Checking feature importance for RF2...

XGB
XGB0 In CV0...

Checking if correct model is loaded...
 XGBClassifier(alpha=0.0003085901759707382, base_score=0.5, booster='gbtree',
              callbacks=None, colsample_bylevel=1, colsample_bynode=1,
              colsample_bytree=0.31595586732894876, early_stopping_rounds=None,
              enable_categorical=False, eta=0.0016131413768891527,
              eval_metric=None, gamma=1.086786493948363e-07, gpu_id=-1,
              grow_policy='lossguide', importance_type=None,
              interaction_constraints='', learning_rate=0.00161314139,
              max_bin=256, max_cat_to_onehot=4, max_delta_step=0, max_depth=3,
              max_leaves=0, min_child_weight=9.912142174935715,
              min_samples_leaf=32, min_samples_split=43, missing=nan,
              monotone_constraints='()', n_estimators=305, n_jobs=1, nthread=1, ...)
['Alanine transaminase (U/L)', 'Albumin (mg/dL)', 'Alcohol', 'Alkaline phosphatase (U/L)', 'Alpha-Fetoprotein (ng/m
L)', 'Ascites degree*', 'Aspartate transaminase (U/L)', 'Chronic Renal Insufficiency', 'Cirrhosis', 'Creatinine (mg/d
L)', 'Diabetes', 'Direct Bilirubin (mg/dL)', 'Encephalopathy degree*', 'Endemic Countries', 'Esophageal Varices', 'Fer
ritin (ng/mL)', 'Gamma glutamyl transferase (U/L)', 'Haemoglobin (g/dL)', 'Hepatitis B Core Antibody', 'Hepatitis B Su
rface Antigen', 'Hepatitis C Virus Antibody', 'International Normalised Ratio*', 'Iron', 'Leukocytes(G/L)', 'Liver Met
astasis', 'Major dimension of nodule (cm)', 'Mean Corpuscular Volume', 'Nonalcoholic Steatohepatitis', 'Number of Nodu
les', 'Oxygen Saturation (%)', 'Packs of cigarets per year', 'Performance Status*', 'Portal Vein Thrombosis', 'Splenom
egaly', 'Symptoms ', 'Total Bilirubin(mg/dL)', 'Total Proteins (g/dL)']
['Alanine transaminase (U/L)', 'Albumin (mg/dL)', 'Alcohol', 'Alkaline phosphatase (U/L)', 'Alpha-Fetoprotein (ng/m
L)', 'Ascites degree*', 'Aspartate transaminase (U/L)', 'Chronic Renal Insufficiency', 'Cirrhosis', 'Creatinine (mg/d
L)', 'Diabetes', 'Direct Bilirubin (mg/dL)', 'Encephalopathy degree*', 'Endemic Countries', 'Esophageal Varices', 'Fer
ritin (ng/mL)', 'Gamma glutamyl transferase (U/L)', 'Haemoglobin (g/dL)', 'Hepatitis B Core Antibody', 'Hepatitis B Su
rface Antigen', 'Hepatitis C Virus Antibody', 'International Normalised Ratio*', 'Iron', 'Leukocytes(G/L)', 'Liver Met
astasis', 'Major dimension of nodule (cm)', 'Mean Corpuscular Volume', 'Nonalcoholic Steatohepatitis', 'Number of Nodu
les', 'Oxygen Saturation (%)', 'Packs of cigarets per year', 'Performance Status*', 'Portal Vein Thrombosis', 'Splenom
egaly', 'Symptoms ', 'Total Bilirubin(mg/dL)', 'Total Proteins (g/dL)']

Checking explainer for XGB0...
<shap.explainers._tree.Tree object at 0x7f9b5595f1c0>

Checking shap values for XGB0...

[[ 0.00033414 -0.00198181  0.         ...  0.          0.
   0.00147086]
 [-0.0001525   0.00104749  0.         ...  0.          0.
   0.00147086]
 [ 0.00033414 -0.00198181  0.         ...  0.          0.
   0.00174883]
 ...
 [ 0.00013163 -0.00198181  0.         ...  0.          0.
  -0.00234208]
 [-0.0001525  -0.00198181  0.         ...  0.          0.
   0.00110088]
 [ 0.0001405   0.00104749  0.         ...  0.          0.
   0.00082496]]

Checking shap plots for XGB0...

Expected value for XGB: 0.00014352845028042793
Summary Plot for SHAP Values in Test Set:

SHAP Bar Plot for SHAP Values Test Set:

SHAP Decision Plot for SHAP Values in Test Set:

SHAP Decision Plot for Single-Prediction in Test Set:
```

```
Checking feature importance for XGB0...

XGB1 In CV1...

Checking if correct model is loaded...
 XGBClassifier(alpha=0.2551087479356069, base_score=0.5, booster='gbtree',
              callbacks=None, colsample_bylevel=1, colsample_bynode=1,
              colsample_bytree=0.7752104293797693, early_stopping_rounds=None,
              enable_categorical=False, eta=0.01045236594154273,
              eval_metric=None, gamma=1.2055493072793457e-05, gpu_id=-1,
              grow_policy='depthwise', importance_type=None,
              interaction_constraints='', learning_rate=0.0104523655,
              max_bin=256, max_cat_to_onehot=4, max_delta_step=0, max_depth=11,
              max_leaves=0, min_child_weight=1.1403861951454617,
              min_samples_leaf=23, min_samples_split=24, missing=nan,
              monotone_constraints='()', n_estimators=372, n_jobs=1, nthread=1, ...)
['Alanine transaminase (U/L)', 'Albumin (mg/dL)', 'Alkaline phosphatase (U/L)', 'Alpha-Fetoprotein (ng/mL)', 'Arterial
Hypertension', 'Ascites degree*', 'Aspartate transaminase (U/L)', 'Cirrhosis', 'Creatinine (mg/dL)', 'Diabetes', 'Dire
ct Bilirubin (mg/dL)', 'Encephalopathy degree*', 'Endemic Countries', 'Ferritin (ng/mL)', 'Gamma glutamyl transferase
 (U/L)', 'Haemoglobin (g/dL)', 'Hemochromatosis', 'Hepatitis B e Antigen', 'Hepatitis C Virus Antibody', 'Internationa
l Normalised Ratio*', 'Iron', 'Leukocytes(G/L)', 'Liver Metastasis', 'Major dimension of nodule (cm)', 'Mean Corpuscul
ar Volume', 'Nonalcoholic Steatohepatitis', 'Number of Nodules', 'Obesity', 'Oxygen Saturation (%)', 'Packs of cigaret
s per year', 'Performance Status*', 'Platelets', 'Portal Hypertension', 'Portal Vein Thrombosis', 'Symptoms ', 'Total
 Bilirubin(mg/dL)']
['Alanine transaminase (U/L)', 'Albumin (mg/dL)', 'Alkaline phosphatase (U/L)', 'Alpha-Fetoprotein (ng/mL)', 'Arterial
Hypertension', 'Ascites degree*', 'Aspartate transaminase (U/L)', 'Cirrhosis', 'Creatinine (mg/dL)', 'Diabetes', 'Dire
ct Bilirubin (mg/dL)', 'Encephalopathy degree*', 'Endemic Countries', 'Ferritin (ng/mL)', 'Gamma glutamyl transferase
 (U/L)', 'Haemoglobin (g/dL)', 'Hemochromatosis', 'Hepatitis B e Antigen', 'Hepatitis C Virus Antibody', 'Internationa
l Normalised Ratio*', 'Iron', 'Leukocytes(G/L)', 'Liver Metastasis', 'Major dimension of nodule (cm)', 'Mean Corpuscul
ar Volume', 'Nonalcoholic Steatohepatitis', 'Number of Nodules', 'Obesity', 'Oxygen Saturation (%)', 'Packs of cigaret
s per year', 'Performance Status*', 'Platelets', 'Portal Hypertension', 'Portal Vein Thrombosis', 'Symptoms ', 'Total
 Bilirubin(mg/dL)']

Checking explainer for XGB1...
<shap.explainers._tree.Tree object at 0x7f9b55b35550>

Checking shap values for XGB1...

[[-4.1213792e-02 -3.3411807e-01  3.1376600e-01 ... -3.7167885e-03
   5.5054866e-02 -5.6627635e-02]
 [ 8.7244794e-02  3.1003144e-03 -1.0110403e-01 ... -8.8498723e-03
  -2.7989864e-01 -7.0158280e-02]
 [-4.6925444e-02  1.0683852e-01  3.3140031e-01 ... -7.7774939e-03
  -2.4780509e-01 -1.7681159e-04]
 ...
 [-1.6688924e-02 -1.4458889e-01  2.3780176e-01 ... -4.2031193e-03
  -2.1513927e-01 -1.9822428e-02]
 [-1.9945810e-02  1.5445979e-01  2.7486852e-01 ...  8.2303239e-03
   7.4652217e-02 -1.7326010e-02]
 [-3.5156224e-02 -2.9176781e-01  7.5178228e-02 ... -5.8821207e-03
   5.9577417e-02 -8.8170484e-02]]

Checking shap plots for XGB1...

Expected value for XGB: 0.40734654664993286
Summary Plot for SHAP Values in Test Set:

SHAP Bar Plot for SHAP Values Test Set:

SHAP Decision Plot for SHAP Values in Test Set:

SHAP Decision Plot for Single-Prediction in Test Set:


Checking feature importance for XGB1...

XGB2 In CV2...

Checking if correct model is loaded...
 XGBClassifier(alpha=2.925626174679448e-07, base_score=0.5, booster='gbtree',
              callbacks=None, colsample_bylevel=1, colsample_bynode=1,
              colsample_bytree=0.10192679583709846, early_stopping_rounds=None,
              enable_categorical=False, eta=0.0002501390367621765,
              eval_metric=None, gamma=4.155123430773271e-05, gpu_id=-1,
              grow_policy='depthwise', importance_type=None,
              interaction_constraints='', learning_rate=0.000250139041,
              max_bin=256, max_cat_to_onehot=4, max_delta_step=0, max_depth=2,
              max_leaves=0, min_child_weight=6.224435947630516,
              min_samples_leaf=10, min_samples_split=45, missing=nan,
              monotone_constraints='()', n_estimators=956, n_jobs=1, nthread=1, ...)
['Albumin (mg/dL)', 'Alcohol', 'Alkaline phosphatase (U/L)', 'Alpha-Fetoprotein (ng/mL)', 'Arterial Hypertension', 'As
cites degree*', 'Aspartate transaminase (U/L)', 'Chronic Renal Insufficiency', 'Cirrhosis', 'Creatinine (mg/dL)', 'Dia
betes', 'Direct Bilirubin (mg/dL)', 'Endemic Countries', 'Ferritin (ng/mL)', 'Gamma glutamyl transferase (U/L)', 'Haem
oglobin (g/dL)', 'Hemochromatosis', 'Hepatitis B Surface Antigen', 'Hepatitis B e Antigen', 'Hepatitis C Virus Antibod
y', 'Human Immunodeficiency Virus', 'International Normalised Ratio*', 'Iron', 'Leukocytes(G/L)', 'Liver Metastasis',
 'Major dimension of nodule (cm)', 'Mean Corpuscular Volume', 'Nonalcoholic Steatohepatitis', 'Number of Nodules', 'Ox
ygen Saturation (%)', 'Packs of cigarets per year', 'Performance Status*', 'Platelets', 'Portal Hypertension', 'Portal
Vein Thrombosis', 'Smoking']
['Albumin (mg/dL)', 'Alcohol', 'Alkaline phosphatase (U/L)', 'Alpha-Fetoprotein (ng/mL)', 'Arterial Hypertension', 'As
cites degree*', 'Aspartate transaminase (U/L)', 'Chronic Renal Insufficiency', 'Cirrhosis', 'Creatinine (mg/dL)', 'Dia
betes', 'Direct Bilirubin (mg/dL)', 'Endemic Countries', 'Ferritin (ng/mL)', 'Gamma glutamyl transferase (U/L)', 'Haem
oglobin (g/dL)', 'Hemochromatosis', 'Hepatitis B Surface Antigen', 'Hepatitis B e Antigen', 'Hepatitis C Virus Antibod
y', 'Human Immunodeficiency Virus', 'International Normalised Ratio*', 'Iron', 'Leukocytes(G/L)', 'Liver Metastasis',
 'Major dimension of nodule (cm)', 'Mean Corpuscular Volume', 'Nonalcoholic Steatohepatitis', 'Number of Nodules', 'Ox
ygen Saturation (%)', 'Packs of cigarets per year', 'Performance Status*', 'Platelets', 'Portal Hypertension', 'Portal
Vein Thrombosis', 'Smoking']

Checking explainer for XGB2...
<shap.explainers._tree.Tree object at 0x7f9b56c2a790>

Checking shap values for XGB2...

[[ 5.08398050e-03  1.09896610e-05 -1.44367898e-02 ... -2.88566196e-04
  -5.45803457e-04  5.12634986e-04]
 [ 4.83505754e-03  2.37361382e-06  1.88300461e-02 ...  5.14981046e-04
  -4.53948451e-04 -9.65057523e-04]
 [ 4.78631770e-03 -3.62699211e-05  1.88307650e-02 ... -2.88566196e-04
  -5.20780042e-04  5.26774325e-04]
 ...
 [ 4.53065475e-03  2.37361382e-06  1.87011994e-02 ... -2.88566196e-04
  -4.90862702e-04 -9.11067938e-04]
 [-1.01444125e-02  2.37361382e-06  9.76637285e-03 ... -2.88566196e-04
  -4.72357147e-04 -8.52823956e-04]
 [ 4.44364175e-03  1.09896610e-05 -2.03168858e-02 ... -2.76739825e-04
  -5.45803457e-04  5.12634986e-04]]

Checking shap plots for XGB2...

Expected value for XGB: -0.0014823254896327853
Summary Plot for SHAP Values in Test Set:

SHAP Bar Plot for SHAP Values Test Set:

SHAP Decision Plot for SHAP Values in Test Set:

SHAP Decision Plot for Single-Prediction in Test Set:


Checking feature importance for XGB2...
```

Run SHAP for Training Sets

Optional

- This runs on training CV Datasets that were partiioned during STREAMLINE
- User can set run_train to 'True' for comparison between training and testing sets

In [ ]:

```
run_force_plots = True # parameter in run_force_plot(); set to True if user wants to display force plots for trained m
run_train = False # user can change to True to run shap values for training sets

if run_train == True:
    for each in datasets:
        print("----------------------------------------")
        print(each)
        print("----------------------------------------")
        full_path = experiment_path+'/' + each

        #Make folder in experiment folder/datafolder to store all shap_values per algorithm/CV combination
        if not os.path.exists(full_path+'/model_evaluation/shap_values/trainResults'):
            os.mkdir(full_path+'/model_evaluation/shap_values/trainResults')

        original_headers = pd.read_csv(full_path+"/exploratory/OriginalFeatureNames.csv",sep=',').columns.values.tolis
        feat_order_map = {feat:i for i, feat in enumerate(original_headers)}
        print(feat_order_map)

        for algorithm in algorithms: #loop through algorithms
            print(abbrev[algorithm])


            for cvCount in range(0,cv_partitions): #loop through cv's
                print('{}{} In CV{}...'.format(abbrev[algorithm], cvCount, cvCount))

                # unpickle and load model
                result_file = full_path+ '/models/pickledModels/' + abbrev[algorithm]+ "_" + str(cvCount)+".pickle"
                file = open(result_file, 'rb')
                model = pickle.load(file)
                file.close()
                print('\nChecking if correct model is loaded...\n', model)


                # Load CV datasets, paths to datasets updates with each iteration
                train_path = experiment_path + train_file_path + 'hcc-data_example_CV_' + str(cvCount) + '_Train.csv'
                test_path = experiment_path + test_file_path + 'hcc-data_example_CV_' + str(cvCount) + '_Test.csv'
                trainX, trainY,testX, testY, train_feat, test_feat = dataPrep(train_path,instance_label,class_label, t


                # shap computation and plots
                explainer = get_explainer(model, abbrev[algorithm], trainX)
                print('\nChecking explainer for {}{}...\n{}'.format(abbrev[algorithm], cvCount, explainer))  # print e
                print('\nChecking shap values for {}{}...\n'.format(abbrev[algorithm], cvCount))
                shap_values = compute_shapValues(model, abbrev[algorithm], explainer, trainX)

                print('\nChecking shap plots for {}{}...\n'.format(abbrev[algorithm], cvCount))
                shap_summary(abbrev[algorithm], train_feat, shap_values, explainer, trainX)


                #save SHAP FI results
                print('\nChecking feature importance for {}{}...\n'.format(abbrev[algorithm], cvCount))
                shap_fi_df = shap_feature_ranking(abbrev[algorithm], shap_values, trainX, train_feat) # can either cho

                filepath = full_path+"/model_evaluation/shap_values/trainResults/"+ abbrev[algorithm] + '_' + str(cvCo
                shap_fi_df.to_csv(filepath, header=True, index=True)


                # only runs force plots if run = True
                if run_force_plots == True:
                    if abbrev[algorithm] in ['NB']:

                        print('\nForce Plot for {}{} SHAP Values in Train Set: \n'.format(abbrev[algorithm], cvCount))
                        shap.force_plot(shap_values, trainX, feature_names=train_feat)

                        print('\nSingle-Prediction Force Plot for {}{} SHAP Values in Train Set: \n'.format(abbrev[alg
                        shap.force_plot(shap_values[42], trainX.iloc[42], feature_names=train_feat, show=False)
    # #                     plt.savefig(full_path+'/model_evaluation/'+abbrev[algorithm]+"_shapFP.png",dpi=300)  FIXME
                        break


                    elif abbrev[algorithm] in ['LR', 'XGB', 'LGB', 'CBG']:  #need to test out LGB and CBG for this

                        print('\nForce Plot for {}{} SHAP Values in Whole Train Set:  \n'.format(abbrev[algorithm], cv
                        shap.force_plot(explainer.expected_value, shap_values, trainX, feature_names=train_feat)

                        print('\nSingle-Prediction Force Plot for {}{} SHAP Values in Train Set: \n'.format(abbrev[alg
                        shap.force_plot(explainer.expected_value, shap_values[42], trainX.iloc[42], feature_names=trai
                        break


                    else:
    #
                        print('\nForce Plot for {}{} SHAP Values from Class 0 in Train Set: \n'.format(abbrev[algorith
                        shap.force_plot(explainer.expected_value[0], shap_values[0], feature_names=train_feat)

                        print('\nForce Plot for {}{} SHAP Values from Class 1 in Train Set: \n'.format(abbrev[algorith
                        shap.force_plot(explainer.expected_value[1], shap_values[1], feature_names=train_feat)
                        break
```

In [ ]:

In [ ]: