Enhancing Patient Care AI For Hepatocellular Carcinoma

Nathaniel Ho (njh2135), Kelly Du (xd2286), Yunchen Jiang (yj2733)

December 16, 2023

Abstract

The project aims to address the challenge of accurately diagnosing hepatocellular carcinoma (HCC) in cirrhotic patients with indeterminate liver nodules. Currently, visual assessment and existing guidelines do not provide reliable differentiation between HCC and other diagnoses, leading to the need for invasive procedures such as liver biopsy or wait and see strategy that can be detrimental to the patient [1]. By leveraging radiomics and artificial intelligence (AI) techniques, this project seeks to deploy a non-invasive diagnostic tool that enhances clinician decision-making, reduces reliance on invasive procedures, and optimizes patient management for improved outcomes.

1 Introduction

In an effort to standardize and improve consensus in the interpretation and reporting of liver computed tomography (CT) scans in patients at risk of hepatocellular carcinoma (HCC), LI-RADS (Liver Imaging Reporting and Data System) was introduced in March 2011 and has been adopted by numerous clinical practices globally. LI-RADS categorizes liver nodules identified on CT scans in high-risk HCC patients into five categories (LI-RADS 1-5), ranging from definitely benign to definitely HCC. The LI-RADS Management Working Group, composed of recognized HCC management experts and radiologists, was convened to assess the management implications associated with the radiological categorization of nodules and their probability of being HCC [1]. In order to get the necessary background information for the project, we referenced several paper [3]. We also reviewed a paper which gave an overview of LI-RADS Guidelines. LI-RADS, introduced in 2011 for standardizing and reporting CT and MRI liver examinations in hepatocellular carcinoma (HCC)-susceptible patients, assigns nodules into five categories; the paper then explores its inception, the initial consensus of its Management Working Group in 2013, and subsequent efforts to reconcile discordance with existing guidelines, aiming to enhance global consensus among physicians managing HCC diagnosis and treatment [4]. We aim to improve clinical decision-making by diagnosing HCC in cirrhotic patients with uncertain liver nodules using quantitative imaging features extracted from triphasic CT scans. We have validated a single-feature signature in a multicenter retrospective cohort for diagnosing HCC in cirrhotic patients with uncertain liver nodules. Artificial intelligence has the potential to assist clinicians by identifying a subgroup of patients at a high risk of HCC.

Key Clinical Points:

1. In cirrhotic patients with visually indeterminate liver nodules, expert visual assessment using current guidelines cannot accurately differentiate HCC from differential diagnoses [1]. Current clinical protocols do not entail biopsy due to procedural risks.

2. Radiomics can be used to non-invasively diagnose HCC in cirrhotic patients with indeterminate liver nodules, which could be leveraged to optimize patient management.Radiomics features contributing the most to a better characterization of visually indeterminate liver nodules include changes in nodule phenotype between arterial and portal venous phases: the ”washout” pattern appraised visually using EASL and EASL guidelines [3].

3. A clinical decision algorithm using radiomics could be applied to reduce the rate of cirrhotic patients requiring liver biopsy (EASL guidelines) or wait-and-see strategy (AASLD guidelines) and therefore improve their management and outcome.

2 Data and Exploratory Data Analysis

2.1 Data Overview

This part outlines the comprehensive data pre-processing steps undertaken to prepare the three datasets for subsequent machine learning analysis.

2.1.1 Training Set: HCC vs. non-HCC (3 Timepoints per Patient)

The training set dataset comprises information crucial for predicting the phenotype, specifically the occurrence of Hepatocellular Carcinoma (HCC) or nonHCC. It encompasses 89 instances of HCC and 21 instances of nonHCC, including 18 benign lesions and 3 malignant lesions. The dataset is organized based on different timepoints, namely 1 (Noncontrast), 4 (Arterial Phase), and 7 (Portal Venous Phase), each providing distinct insights into the patients' conditions. Notably, delta features are computed, creating three sets: 4-1, 7-4, and 7-1. These delta features could potentially reveal valuable information about the evolution of the patients' conditions over time.

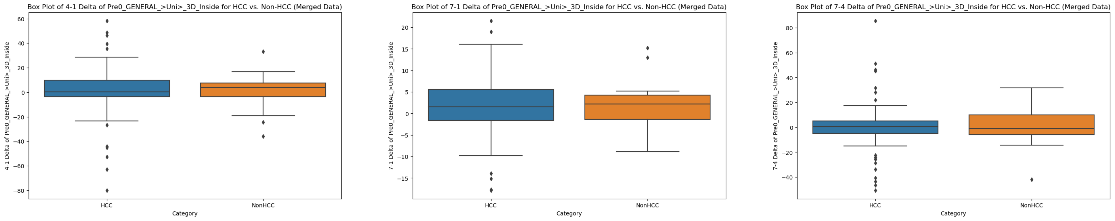


Figure 1: Example of Delta features of Pre0_GENERAL_>Uni>_3D_Inside' feature between HCC and Non-HCC.

2.1.2 All Phenotypes: Portal Venous Phase

The All Phenotypes dataset focuses on a specific time point, namely 7 (Portal Venous Phase), where all patients are categorized as "nonHCC." The primary goal of this dataset is to identify features during the portal venous phase that can effectively differentiate between HCC and nonHCC. This dataset is pivotal in contributing valuable insights into the characteristics of nonHCC cases, providing a foundation for developing a robust predictive model when combined with the training set.

2.1.3 Advanced HCC: Portal Venous Phase

In contrast, the Advanced HCC dataset is exclusively composed of patients at the portal venous phase (timepoint 7), and all patients are labeled as "HCC." The role of this dataset is to pinpoint features during the portal venous phase that can effectively differentiate between HCC and non-HCC. By focusing on the advanced cases of HCC, this dataset contributes essential information to enhance the predictive power of the model. Combining insights from the Advanced HCC dataset with the other datasets allows for a comprehensive understanding of the key differentiating factors between HCC and non-HCC cases, facilitating more accurate predictions in the training set.

2.2 Dimensionality Reduction

The classification of indeterminate liver nodule is an unbalanced problem with high dimensionality since the predominant diagnosis is HCC. In this project, we address these problems through a comprehensive data preprocessing pipeline. The focus is on creating a binary target variable, mitigating class imbalance using Synthetic Minority Over-sampling Technique (SMOTE), and employing Principal Component Analysis (PCA) for dimensionality reduction.

2.3 Binary Target Variable

The dataset originates from portal venous phase imaging, systematically collected from 27 institutions spanning April 2010 to December 2015. A meticulous retrospective analysis was conducted on 178 patients meeting specific criteria, including cirrhosis and the presence of indeterminate liver nodules. We define a binary target variable named 'Category', where the value is 1 if the original category is 'HCC' (Hepatocellular Carcinoma) and 0 otherwise. This binary variable lays the foundation for subsequent classification tasks.

2.4 Data Split and Resampling

The dataset is split into training and testing sets. The class imbalance is addressed through SMOTE, synthesizing instances of the minority class for a more balanced representation in the training set.

2.4.1 Principal Component Analysis

Subsequently, PCA was performed on the resampled and standardized training data to reduce dimensionality while retaining essential information. A total of 30 principal components were chosen based on their ability to capture the dataset's variability effectively. The feature weights obtained from PCA were visualized in Figure 2, where each bar represented the absolute weight of a feature in a principal component.

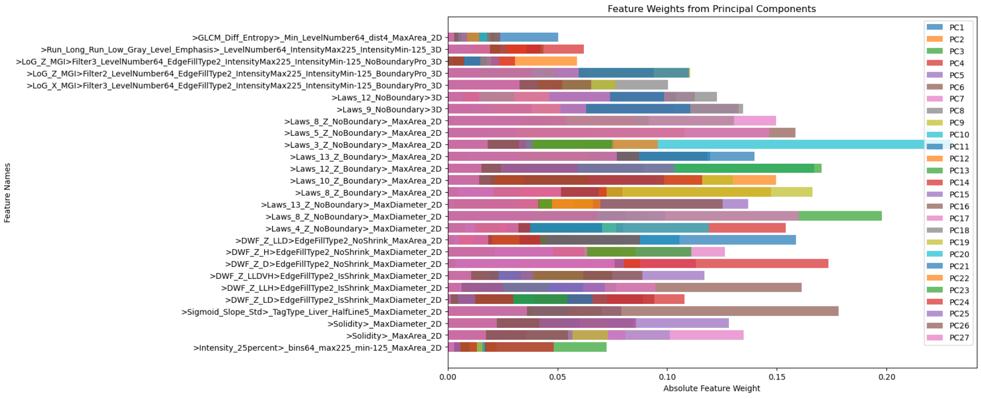


Figure 2: Feature weights obtained from PCA

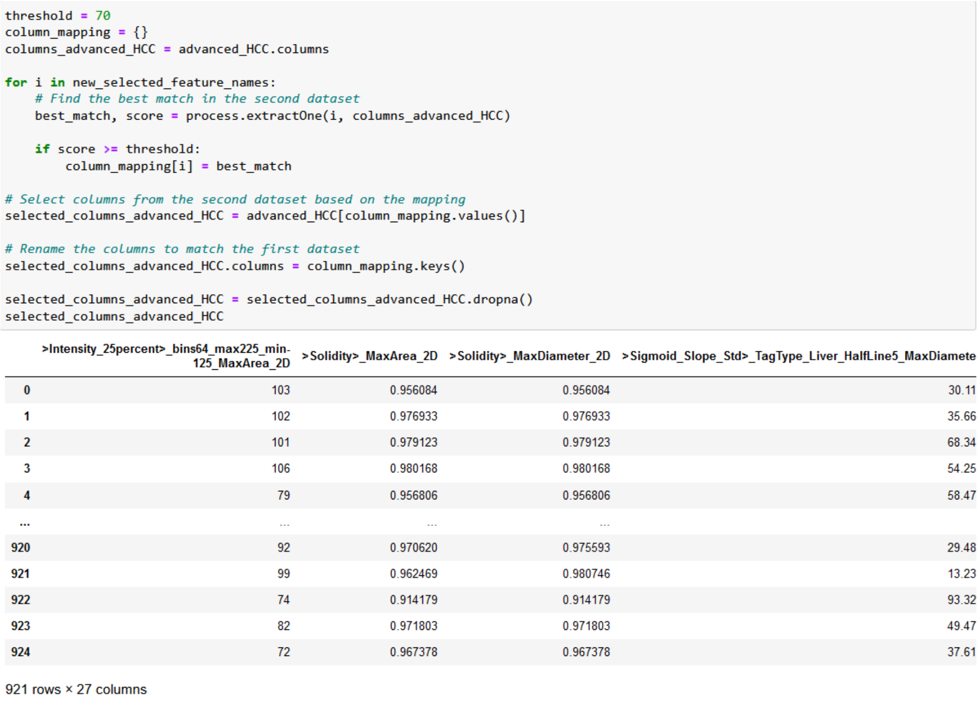
We initially set 'num_components=30'; however, only 27 unique feature indices were obtained. This indicates that there are fewer than 30 unique features with the highest weights across all principal components, which means that some features have identical weights in multiple principal components.

2.5 Feature Matching and Selection

We integrate information from two distinct datasets, advanced_HCC and AllPhenotypes, by selecting relevant features that align with those identified in the primary training dataset. This process involves a fuzzy string-matching approach to find the best-matching columns between the datasets.

2.5.1 Feature Matching in advanced_HCC dataset

Fuzzy string matching was employed to find the best match for each selected feature in the advanced_HCC dataset. A matching threshold of 70 was set to ensure a reasonable degree of similarity. The selected columns from advanced_HCC were then mapped to match the features from the primary training dataset. The resulting dataset, named selected_columns_advanced_HCC, retained 27 columns after cleaning.



```
threshold = 70
column_mapping = {}
columns_advanced_HCC = advanced_HCC.columns

for i in new_selected_feature_names:
    # Find the best match in the second dataset
    best_match, score = process.extractOne(i, columns_advanced_HCC)

    if score >= threshold:
        column_mapping[i] = best_match

# Select columns from the second dataset based on the mapping
selected_columns_advanced_HCC = advanced_HCC[column_mapping.values()]

# Rename the columns to match the first dataset
selected_columns_advanced_HCC.columns = column_mapping.keys()

selected_columns_advanced_HCC = selected_columns_advanced_HCC.dropna()
selected_columns_advanced_HCC
```

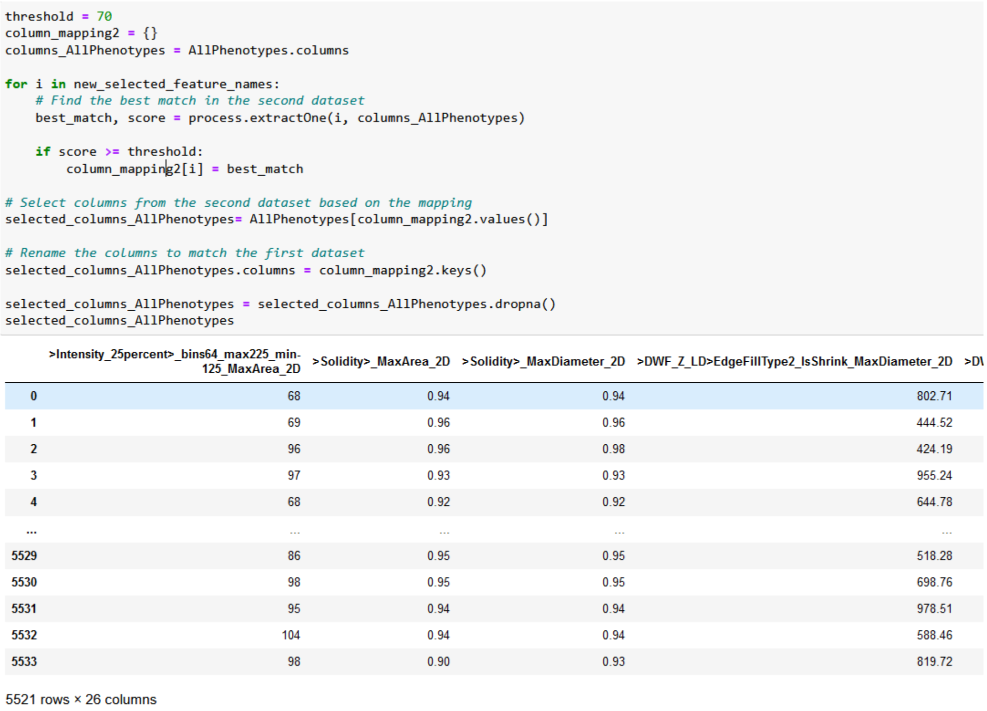
	>Intensity_25percent>_bins64_max225_min-125_MaxArea_2D	>Solidity>_MaxArea_2D	>Solidity>_MaxDiameter_2D	>Sigmoid_Slope_Std>_TagType_Liver_HalfLine5_MaxDiamete
0	103	0.956084	0.956084	30.11
1	102	0.976933	0.976933	35.66
2	101	0.979123	0.979123	68.34
3	106	0.980168	0.980168	54.25
4	79	0.956806	0.956806	58.47
...	...	...	...	
920	92	0.970620	0.975593	29.48
921	99	0.962469	0.980746	13.23
922	74	0.914179	0.914179	93.32
923	82	0.971803	0.971803	49.47
924	72	0.967378	0.967378	37.61

921 rows × 27 columns

Figure 3: 27 features found in Advanced HCC

2.5.2 Feature Matching in AllPhenotypes Dataset

We use the same approach for the AllPhenotypes dataset. However, the resulting dataset, 'selected_columns_AllPhenotypes', only retained 26 columns after cleaning.



```
threshold = 70
column_mapping2 = {}
columns_AllPhenotypes = AllPhenotypes.columns

for i in new_selected_feature_names:
    # Find the best match in the second dataset
    best_match, score = process.extractOne(i, columns_AllPhenotypes)

    if score >= threshold:
        column_mapping2[i] = best_match

# Select columns from the second dataset based on the mapping
selected_columns_AllPhenotypes= AllPhenotypes[column_mapping2.values()]

# Rename the columns to match the first dataset
selected_columns_AllPhenotypes.columns = column_mapping2.keys()

selected_columns_AllPhenotypes = selected_columns_AllPhenotypes.dropna()
selected_columns_AllPhenotypes
```

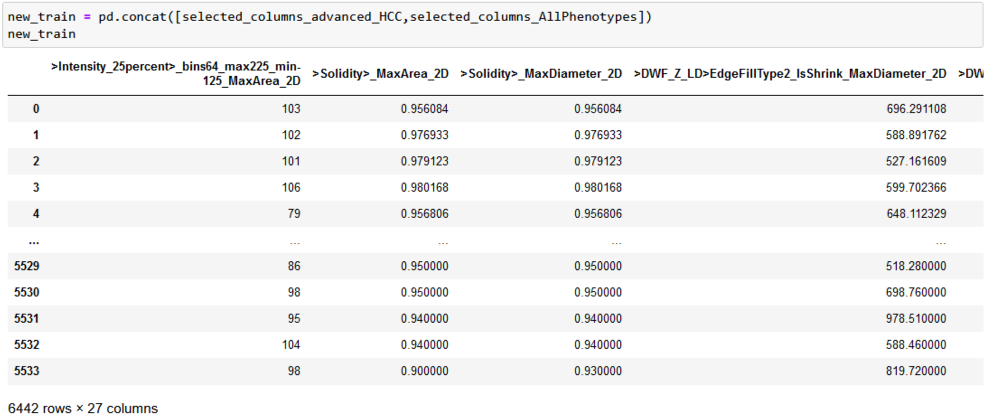
	>Intensity_25percent>_bins64_max225_min-125_MaxArea_2D	>Solidity>_MaxArea_2D	>Solidity>_MaxDiameter_2D	>DWF_Z_LD>EdgeFillType2_IsShrink_MaxDiameter_2D
0	68	0.94	0.94	802.71
1	69	0.96	0.96	444.52
2	96	0.96	0.98	424.19
3	97	0.93	0.93	955.24
4	68	0.92	0.92	644.78
...	...	...	...	...
5529	86	0.95	0.95	518.28
5530	98	0.95	0.95	698.76
5531	95	0.94	0.94	978.51
5532	104	0.94	0.94	588.46
5533	98	0.90	0.93	819.72

5521 rows × 26 columns

Figure 4: 26 features found in AllPhenotypes

2.5.3 Dateset Integration

Since not all 27 selected features could be found across both datasets, we decide to utilize the 26 common features available in the AllPhenotypes dataset. The feature, '>Sigmoid_Slope_Std>_TagType_Liver_HalfLine5_MaxDiameter_2D', is dropped. The final integrated dataset, denoted as 'new_train', is created by concatenating the common features across both datasets. The resulting new_train dataset contains 26 features and includes information from both advanced_HCC and AllPhenotypes datasets. The integrated dataset is used in the development and evaluation of predictive models of this research.



```
new_train = pd.concat([selected_columns_advanced_HCC,selected_columns_AllPhenotypes])
new_train
```

	>Intensity_25percent>_bins64_max225_min-125_MaxArea_2D	>Solidity>_MaxArea_2D	>Solidity>_MaxDiameter_2D	>DWF_Z_LD>EdgeFillType2_IsShrink_MaxDiameter_2D
0	103	0.956084	0.956084	696.291108
1	102	0.976933	0.976933	588.891762
2	101	0.979123	0.979123	527.161609
3	106	0.980168	0.980168	599.702366
4	79	0.956806	0.956806	648.112329
...	...	...	...	...
5529	86	0.950000	0.950000	518.280000
5530	98	0.950000	0.950000	698.760000
5531	95	0.940000	0.940000	978.510000
5532	104	0.940000	0.940000	588.460000
5533	98	0.900000	0.930000	819.720000

6442 rows × 27 columns

Figure 5: New training set. Note that 27 columns contain 26 features and 1 binary target variable

3 Model Building

3.1 General Steps for Model

1. Using the 27 features obtained from the PCA done in EDA/data cleaning, we piped those input features into each model. The data from these features comes from timepoint 7, the venous phase of determining whether a patient has HCC or not; this is considered our "training" dataset.

2. A classification report from running these 27 features into the model of choice is generated, showing initial accuracy, precision, and recall.

3. We then train the same model type on slightly different features (26 features from the PCA model rather than the 27) from a combination of both the Advanced_HCC dataset and the phenotype dataset. The reason why a particular feature was dropped was because we couldn't find the corresponding feature name within either of these datasets. The datasets in question are from the same timepoint as the data from the initial training dataset.

4. Another classification report is generated for these 26 input features, again showing initial accuracy, precision, and recall.

5. We then try out different types of sampling to overcome the imbalanced dataset. In this particular pipeline, we utilize SMOTE as well as stratified K-Folds, and predict/validate our results using the combined dataset above. We then print out the accuracy of the model after utilizing each of these resampling strategies.

6. Finally, we utilize hyperparameter tuning (usually GridSearchCV) to optimize each model, and generate a model with parameters set for highest accuracy, and also generate the accuracy itself as output.

We then move on to a brief overview of each model, its basic mechanisms, and their overall results.

3.2 Random Forest Classifier

A random forest classifier is an ensemble learning method which extends the ML concept of decision trees by relying on multiple versions of this model to create an aggregate prediction result. Each tree acts independently and generates its prediction. Through a process called bagging, where different subsets of the data are used to train each tree, and random feature selection, these trees collectively create a robust model. The final prediction from the random forest is determined by aggregating the individual predictions of each tree, often resulting in enhanced accuracy, generalization, and robustness compared to a single decision tree model.

When training the Random Forest Classifier on our initial training set, and then validating it on the same dataset, we get the following classification results:

Accuracy: 0.8571

	Precision	**Recall**	**F1-Score**	**Support**
Class 0	0.91	0.77	0.83	13
Class 1	0.82	0.93	0.87	15
Accuracy			0.86	28
Macro Avg	0.87	0.85	0.85	28
Weighted Avg	0.86	0.86	0.86	28

Table 1: Random Forest Classification Metrics w/ Train and Val Set

We then move on to the next step, which is training and validating over this dataset while merging data from the phenotypes as well. The result was highly accurate, with an average accuracy of 0.8929.

	Precision	**Recall**	**F1-Score**	**Support**
Class 0	1.00	0.77	0.87	13
Class 1	0.83	1.00	0.91	15
Accuracy			0.89	28
Macro Avg	0.92	0.88	0.89	28
Weighted Avg	0.91	0.89	0.89	28

Table 2: Random Forest Classification Metrics w/ Phenotype Data

Finally, we attempt to increase the accuracy in two ways: first, we use SMOTE (Synthetic Minority Over-sampling Technique) since our data is heavily skewed towards non-HCC patients. We also use hyperparameter tuning separately to see if it yielded better results. The first table represents the results from the SMOTE process, followed by hyperparameter tuning:

	Precision	**Recall**	**F1-Score**	**Support**
Class 0	1.00	1.00	1.00	5521
Class 1	1.00	1.00	1.00	5521
Accuracy			1.00	11042
Macro Avg	1.00	1.00	1.00	11042
Weighted Avg	1.00	1.00	1.00	11042

Table 3: Random Forest Classification Metrics w/ SMOTE

As you can see, we get extreme overfitting when we balance the data using SMOTE; therefore, if we were to move forward with this machine learning model, we would

	Precision	Recall	F1-Score	Support
Class 0	1.00	0.69	0.82	13
Class 1	0.79	1.00	0.88	15
Accuracy			0.86	28
Macro Avg	0.89	0.85	0.85	28
Weighted Avg	0.89	0.86	0.85	28

Table 4: Random Forest Classification Metrics w/ K-Folds

have to utilize alternative sampling strategies to get realistic results. As for the hyperparameter tuning, we do see unusually high number in precision and recall, but this can be remedied by testing a wider variety of options to tune within the model.

3.3 KNN Classifier

A KNN Classifier is a classifier that utilizes each potential HCC node as a point in an n-dimensional space, where n is 26. Because we have five different categories that define HCC vs non-HCC, there are five clusters within this KNN model. As we did with the random forest classifier, we first train and validate this model with the data at timepoint 7. The full results of the training/validation dataset are listed below:

Accuracy: 0.6786

	Precision	Recall	F1-Score	Support
Class 0	0.59	1.00	0.74	13
Class 1	1.00	0.40	0.57	15
Accuracy			0.68	28
Macro Avg	0.80	0.70	0.66	28
Weighted Avg	0.81	0.68	0.65	28

Table 5: KNN Classification Metrics w/ Train and Val

As we did before with Random Forest, we also plugged in the dataset with phenotype data. The result was a lower accuracy score that was only slightly better than random chance: Accuracy: 0.5037 We then move on to SMOTE. SMOTE only increased the accuracy slightly, so we also tried to use K-folds cross-validation, using five folds, as well as hyperparameter tuning using GridSearchCV. The results of these different methods are shown below:

As you can see, SMOTE produces only marginally better results; on the other hand, stratified K-Folds performed much better, while GridSearchCV yielded hyperparameters that made the model more accurate; however, additional tuning of the model is necessary before we reach higher, more acceptable levels of accuracy.

	Precision	**Recall**	**F1-Score**	**Support**
Class 0	0.86	0.50	0.63	5521
Class 1	0.15	0.51	0.23	921
Accuracy			0.50	6442
Macro Avg	0.50	0.51	0.43	6442
Weighted Avg	0.76	0.50	0.58	6442

Table 6: KNN Classification Metrics w/ Phenotype Data

Mean Accuracy	
SMOTE Pre-processing	0.5093
Stratified K-Folds	0.821
GridSearchCV	0.75

Table 7: KNN Classification Metrics w/ Diff. Processing Techniques

3.4 Logistic Regression

Logistic regression is a statistical technique primarily used for binary classification tasks where the outcome is categorical with two possibilities, which in this case are either HCC-positive or HCC-negative. Despite having ”regression” in the name, it is actually a classification approach, and has historically been used in healthcare for disease prevention, hence why we try to extend its use case to our project. It leverages the logistic (or sigmoid) function to transform input values into probabilities ranging from 0 to 1, with a decision boundary or ”tipping point” of 0.5. The estimation of model parameters is done through maximum likelihood estimation, aiming to maximize the likelihood of observing the given data. As per the previous two iterations of models we first run the training/validation dataset on this model, which produced the following results:

Accuracy: 0.5000

	Precision	**Recall**	**F1-Score**	**Support**
Class 0	0.48	0.77	0.59	13
Class 1	0.57	0.27	0.36	15
Accuracy			0.50	28
Macro Avg	0.52	0.52	0.48	28
Weighted Avg	0.53	0.50	0.47	628

Table 8: Logistic Regression w/ Train and Val

As you can see, logistic regression doesn’t perform much better than random chance. To affirm this conclusion, we incorporated the phenotype data as well, which yielded the following results:

	Precision	Recall	F1-Score	Support
Class 0	0.50	0.69	0.58	13
Class 1	0.60	0.40	0.48	15
Accuracy			0.54	28
Macro Avg	0.55	0.55	0.53	28
Weighted Avg	0.55	0.54	0.53	28

Table 9: Logistic Classification Metrics w/ Phenotype Data

Even though there is a three hundred basis point increase in the accuracy, the model still performs poorly overall. If we attempt the same data cleaning techniques as with KNN, we get the following results:

Mean Accuracy	
SMOTE Pre-processing	0.5344
Stratified K-Folds	0.7125
GridSearchCV	0.5357

Table 10: Logistic Classification Metrics w/ Diff. Processing Techniques

From the table above, we can see that stratified k-folds has the highest accuracy compared to other methods. However, all three are still weak relative to our other two models; therefore this model might not be the best fit for our given data.

4 Conclusions and Future Work

The data processing pipeline for predicting Hepatocellular Carcinoma (HCC) involves a meticulous and multifaceted approach. The training set, with 89 instances of HCC and 21 instances of non-HCC across three timepoints, is enriched by delta features and complemented by specialized datasets focusing on portal venous phase imaging. Challenges such as class imbalance and high dimensionality are systematically addressed by introducing a binary target variable ('Category'), employing Synthetic Minority Over-sampling Technique (SMOTE) for class balancing, and implementing Principal Component Analysis (PCA) for dimensionality reduction. The dataset, sourced from 27 institutions spanning 2010 to 2015, undergoes retrospective analysis, focusing on 178 patients meeting specific criteria. The integrated dataset, 'new_train,' emerges from a careful fusion of insights from Advanced HCC and All Phenotypes datasets using fuzzy string matching. This 26-feature dataset is employed in developing and evaluating predictive models, seeking to enhance accuracy in distinguishing HCC and non-HCC cases based on portal venous phase imaging data. The comprehensive preprocessing pipeline not only tackles inherent challenges but also aims to reveal nuanced temporal and phenotypic aspects of Hepatocellular.

For our model, the initial phase involves Principal Component Analysis (PCA) applied to the venous phase data at timepoint 7, resulting in the extraction of 27 features. These features serve as inputs to three distinct models: Random Forest Classifier, KNN Classifier, and Logistic Regression. The Random Forest Classifier, when trained on this dataset, demonstrates promising accuracy, precision, and recall metrics. The subsequent refinement of the feature set incorporates data from both the Advanced_HCC and phenotype datasets, albeit with the exclusion of one feature due to unavailability. However, an unexpected challenge arises when employing Synthetic Minority Over-sampling Technique (SMOTE) to address class imbalance, leading to extreme overfitting. This prompts a critical evaluation of alternative sampling strategies. The KNN Classifier and Logistic Regression models, despite exhibiting varying degrees of success, benefit from additional techniques such as stratified K-Folds for cross-validation and hyperparameter tuning via GridSearchCV. The Logistic Regression model, while showing improvement, remains relatively weak compared to the other models.

Recommendations for further enhancement include a comprehensive exploration of diverse sampling techniques and continued fine-tuning of model parameters. This iterative approach ensures a thorough understanding of the dataset and the selection of the most effective strategies for achieving accurate HCC classification.

References

[1] Mokrane FZ, Lu L, Vavasseur A, Otal P, Peron JM, Luk L, Yang H, Ammari S, Saenger Y, Rousseau H, Zhao B, Schwartz LH, Dercle L. Radiomics machine-learning signature for diagnosis of hepatocellular carcinoma in cirrhotic patients with indeterminate liver nodules. Eur Radiol. 2020 Jan;30(1):558-570. doi: 10.1007/s00330-019-06347-w. Epub 2019 Aug 23. PMID: 31444598.

[2] Dercle L, Lu L, Lichtenstein P, Yang H, Wang D, Zhu J, Wu F, Piessevaux H, Schwartz LH, Zhao B. Impact of Variability in Portal Venous Phase Acquisition Timing in Tumor Density Measurement and Treatment Response Assessment: Metastatic Colorectal Cancer as a Paradigm. JCO Clin Cancer Inform. 2017 Nov;1:1-8. doi: 10.1200/CCI.17.00108. PMID: 30657405; PMCID: PMC6874047.

[3] Eche T, Schwartz LH, Mokrane FZ, Dercle L. Toward Generalizability in the Deployment of Artificial Intelligence in Radiology: Role of Computation Stress Testing to Overcome Underspecification. Radiol Artif Intell. 2021 Oct 27;3(6):e210097. doi: 10.1148/ryai.2021210097. PMID: 34870222; PMCID: PMC8637230.

[4] Mitchell, D.G., Bruix, J., Sherman, M. and Sirlin, C.B. (2015), LI-RADS (Liver Imaging Reporting and Data System): Summary, discussion, and consensus of the LI-RADS Management Working Group and future directions. Hepatology, 61: 1056-1065. https://doi.org/10.1002/hep.27304