Hepatitis C Machine Learning Pipeline Technical Report

Psallidas Kyriakos

[1]Department of Computer Science and Telecommunications, National and Kapodistrian University of Athens

Abstract

This report presents a machine-learning pipeline developed for hepatitis C virus (HCV) prediction. The aim is to assess the performance of different classification algorithms in the task at hand and reliably select and optimize the hyperparameters of the best-performing one. The baseline pipeline designed for HCV prediction consists of three main stages: data preprocessing, model evaluation, and final model selection and tuning. The model evaluation stage utilizes a developed custom class called ***ClassifierCV****, compatible with the sci-kit learn API that performs classifier algorithm evaluation using nested cross-validation trials and Bayesian hyperparameter tuning with optuna. The performance of several classification algorithms was evaluated, including linear, rbf, sigmoid, poly-support vector machines (SVMs), K nearest neighbours (KNN), linear discriminant analysis (LDA), Gaussian naive Bayes (GNB) and lasso,ridge, elastic logistic regression (LR). The results indicate that SVMs with a liner kernel performed best on the 12-D feature space.*

Introduction

Hepatitis C virus (HCV) is a viral infection that results in liver inflammation. While some HCV infections may be short-term, asymptomatic, and not life-threatening, the majority (approximately 70%) progress to chronic infection, which progressively increases the risk of developing cirrhosis [1], a serious and life-threatening condition characterized by the formation of scar tissue in the liver. Due to the crucial importance of early detection and/or clinical stage of HCV, Machine learning algorithms that are able to uncover patterns in laboratory data that may not be apparent to humans are a valuable non-invasive tool to assist medical personnel.

In this context, the aim of this report is to present a machine-learning pipeline developed for HCV prediction to assess the performance of different classification algorithms in the task at hand and reliably select and optimize the hyper-parameters of the best-performing one.

Methodologies

Data-set Description

This study utilizes a dataset consisting of 204 samples, comprising data on hepatitis C patients and healthy blood donors. Each sample corresponds to a patient's ID, and 12 features are available for each donor. To visualize the data-set, a t-SNE projection based on the minimization of the Kullback-Leibler divergence between the estimated t-distribution kernel density joint probability function of the data points in the original high-dimensional space and a lower-dimensional space was utilized (figure 3(a)). The first two features of the dataset capture the age and sex of the donor, while the remaining features represent the levels of various molecules obtained from blood chemistry results, specifically from liver function tests as presented in (table 1). Concerning the target labels, the blood donors are binary classified into healthy (label=0), or HCV positive (label=1). It's important to note that as with most medical datasets, the samples for the labels are imbalanced with the HCV-positive class being the minority class and accounting for 50% of the number of samples of the majority HCV-negative class.

Table 1: *Features*

Feature	Category
Age	Discrete
Sex	Binary
ALB (albumin)	Continuous
ALP (Alkaline phosphatase)	Continuous
ALT (Alanine transaminase)	Continuous
AST (Aspartate transaminase)	Continuous
BIL (Bilirubin)	Continuous
CHE (Cholinesterase)	Continuous
CHOL (Cholesterol)	Continuous
CREA (Creatinine)	Continuous
GGT (Gamma-glutamyltransferase)	Continuous
PROT (Total Protein)	Continuous

Many of the laboratory results featured in the data set are measured in different scales, either units per liter (U/L), grams per deciliter (g/dl) or milligrams per deciliter (mg/dL). [2]. Additionally, the majority of them include outliers, observations above or below 150% of the interquartile range (IQR) from the first

and the third quartile respectively (figure 1).

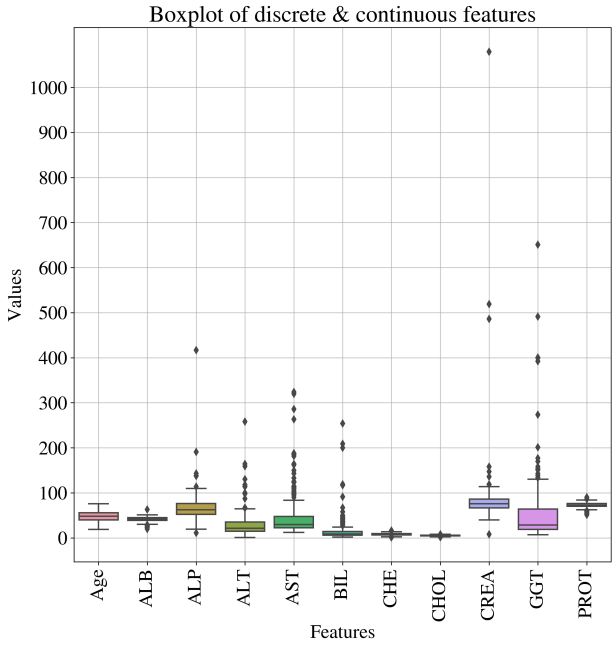


Figure 1: *Box-plot of the un-processed features excluding binary features*

Base Pipeline

Machine learning pipelines provide several significant advantages, such as reproducibility, scalability, and efficiency, by presenting a concise and interpretable way to run machine learning methods. The baseline pipeline designed for Hepatitis C prediction consists of three main stages: data preprocessing, model evaluation, final Model Selection & tuning.

Data Preprocessing with the inclusion of **outlier removal** by removing observations that lie 1.5 time the value of the Interquartile range (IQR) below or above the first (Q_1) or third (Q_3) quarterlies respectively was investigated, but it was deemed unsuitable because it results in losing 69% of the HCV-positive class and only 12.5% of the HCV-negative class. This observation is reasonable in the context of medical laboratory data, where a large deviation from the normal range is often indicative of pathology. Regarding **feature scaling**, both standardization and normalization methods were examined, but ultimately normalization was chosen as illustrated in the "Preprocessing" block in (figure 4). This decision was made because normalization bounds the feature values between $0-1$, which, when combined with the binary values in the sex feature $\in \{0,1\}$, effectively differentiates between the sexes for classification models. This clear distinction is evident in (figures 3(b) and 3(c)). Of course, this places significantly more weight on one specific feature and can introduce artificial bias, thus should be reasonably justified. The logic behind this choice is the fact that most often laboratory result expected levels differ between males and females. This claim is validated specifically for liver function test in [2].

For Model Evaluation , a custom class called *ClassifierCV* was created. This class is fully compatible with the sci-kit learn API and is designed to perform classifier algorithm evaluation using nested cross-validation trials and Bayesian hyper-parameter tuning inside the inner cross-validation loop with the optuna library [3, 4]. The "Model Evaluation" block in (figure 4) showcases this approach. By combining nested cross-validation and Bayesian hyper-parameter tuning with a TPE (Tree-structured Parzen Estimator), *ClassifierCV* enables a comprehensive evaluation of the model's performance and ensures that the model is not over-fitting on a specific train/test split. Bayesian hyperparameter tuning was chosen over Grid or Random based methods since it allows for informative tuning. *ClassifierCV* accepts several classifier categories, including K nearest neighbours (KNN), Gaussian naive Bayes (GNB), lasso logistic regression (LR), ridge LR, elastic LR, linear discriminant analysis (LDA), polynomial kernel support vector machines (SVM), radial basis function kernel SVM, sigmoid kernel SVM, and linear kernel SVM. The F_{beta} score is used as the metric to be maximized by optuna's objective function inside the inner cross-validation loop and the model with the hyperparameters that produced the highest F_{beta} score is evaluated in the outer cross-validation loop each time.

$$F_{beta} = (1 + beta^2)\frac{precision \times recall}{(beta^2 \times precision) + recall}$$

The $beta(deault = 1)$ parameter is a class argument and can be set by the user. For Hepatitis C prediction the most important task is to minimize the number of blood donors that are HCV positive but are classified as HCV negative (FN). For this reason, for HVC prediction, we have set $beta = 2$ to prioritize a better $Recall = \frac{TP}{TP+FN}$ over $Precision = \frac{TP}{TP+FP}$. Another important parameter of *ClassifierCV* is the class weight, which determines the weight assigned to each class in applicable algorithms such as LR and SVM classifiers. By default, the weights are set to equal $\{1 : 1, 0 : 1\}$. In the case of the Hepatitis C prediction pipeline, the weight for the minority HCV-positive class is set to double compared to the negative class $\{1 : 2, 0 : 1\}$ following the cardinality of each class in the dataset, to introduce further cost-sensitive learning to the imbalanced dataset. The remaining parameters correspond to the number of cross-validation trials, number of outer stratified cross-validation splits and

number of inner cross-validation trials. We have utilized 10 uniquely seeded trials of 5 outer and 3 inner splits, for a total of 50 nested cross-validation loops per classifier evaluated. For each one of the 50 loops The best classifier parameters from the inner loop and the F_1, F_{beta}, *ROC AUC*,*Balanced Accuracy*, *Mathews Coefficient*, *Precision*, *Recall*, *Specificity*, *Negative Predictive Values* are collected and stored in a data frame.

Final Model Selection & Tuning is accomplished by picking the classifier algorithm with the highest *Matthews correlation* (MCC) mean score across the 50 nested cross validation loops.

$$MCC = \frac{TP \times TN - FP \times FN}{\sqrt{(TP+FP)(TP+FN)(TN+FP)(TN+FN)}}$$

The MCC metric is chosen as the selection criterion because it yields high scores only when the classifier accurately predicts both positive and negative data instances. This makes it insensitive to class balance. MCC ranges between -1 and +1, where -1 and +1 denote perfect misclassification and perfect classification, respectively. A score of 0 indicates the classifier is performing randomly [5].

Once the optimal classifier algorithm is identified, it is fitted to the entire dataset's features X. Bayesian hyperparameter seeded tuning is performed using a non-nested 5-fold cross-validation loop of *ClassifierCV* to select the best hyperparameters for the classifier with optuna's TPE (Tree-structured Parzen Estimator) algorithm.

Feature Selection Methods

To select the top five features and assess the selection's impact on the performance of the baseline pipeline, we utilized two methods that assign importance to features with respect to the target label. The first method involves selecting the five features with the highest **ANOVA F-score**, which measures the ratio of the variance of each feature between class labels to the variance within each feature group. As ANOVA F-score relies on linear relationships, we also employed a second selection method based on entropy, which does not assume any specific relationship. This method involves calculating the **Mutual Information** between each feature X and label Y, using the formula

$$I(X;Y) = H(X) - H(X|Y)$$

Mutual Information measures the reduction in uncertainty about the label by knowing each of the features.

Results & Discussion

Classifier algorithm evaluation

The results from the model evaluation pipeline stage, presented in (figure 5) and discussed below, showcase that linear-SVC outperformed other classifiers in binary classification between healthy blood donors and HVC-positive donors. They achieved the highest mean and one of the lowest standard deviations across all classification metric scores, except for specificity, as demonstrated in figure 5(b)). Specifically, linear SVC achieved a mean MCC score of 0.826 with a standard deviation of 0.1 over 50 CV loops. This represents an improvement of $\approx 14\%$ in mean performance compared to the GNB baseline classifier, which had a mean MCC score of 0.717. The improvement in classifier prediction standard deviation, although the second best, was minor between the linear-SVC and the baseline GNB classifier ($0.1 \sim 0.108$). However, It is important to note that linear SVCs exhibit a considerable left-skewed distribution for most classification metrics, particularly recall, which we deem highly essential for the current task. The majority of values tend to cluster towards the positive side, as evidenced by the box-plot distribution of all classification metrics (see Figure 5(a)) and the MCC metric specifically (see Figure 2). This means that the mean is pulled to the left by the presence of a few very low values on the left side of the distribution, while the bulk of the score values is located towards the right side, thus the mean underestimates the most common values in the linear-SVC MCC score distribution and we can expect better MCC scores.

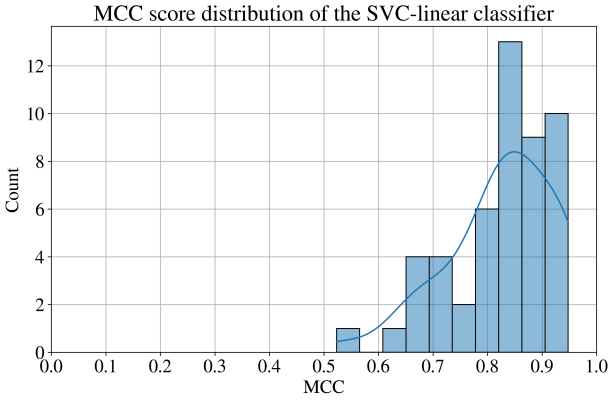


Figure 2: *Histogram of 10 bins of the SVC-linear MCC score over 50 CV trials*

Ridge LR had the second-best MCC score of 0.812 and a standard deviation similar as to the linear SVC ($0.99 \sim 0.1$). Lasso and elastic LR performed worse than ridge LR with 0.794, 0.810 MCC scores respectively and similar variance to ridge LR. Thus we

can conclude that prioritizing small via $l2$ regularization rather than sparse feature weight vectors via $l1$ proves to be a more effective regularization technique for this dataset.

The polynomial kernel SVC, also resulted in an MCC score of 0.812, but with a lower standard deviation = 0.90 than ridge LR and liner SVC. The remaining SVC classifiers, with radial basis function and sigmoid kernels, performed significantly worse with $0.777, 0.734$ MCC scores respectively and the highest standard deviations of all classifiers 0.152, 0.240. This observation, along with the higher LR and liner-SVC scores suggests that the decision boundary between the classifiers is linear in nature, or can be approximated as such via regularization.

Regarding KNN and LDA classifiers, they result in the two lowest MCC scores of 0.610 and 0.707 respectively with standard deviations of 0.121 and 0.127. The KNN algorithm's low performance can be traced back to several factors, including class imbalance, the presence of outliers, and the curse of dimensionality, as KNN relies solely on distance-based classification. On the other hand, LDA utilizes a projection that preserves the maximum separability between classes and is less affected by the curse of dimensionality. However, like KNN, it assumes balanced classes, which likely accounts for its lower performance.

After examining all the classifiers, we have decided to retain our initial choice of the linear kernel SVC as the most effective classifier for HVC prediction. While the SVC with a polynomial kernel was a close second, we determined that the improvement of a 0.010 decrease in standard deviation was not worth sacrificing the higher MCC and Recall scores of 0.14 and 0.15 respectively. Additionally, we took into consideration the significant left skewness in the Recall score distribution for the SVC classifier, which is the most crucial score for a model deployed in a diagnostic setting to minimize misclassification of HVC-positive patients as healthy.
The final model after fitting an SVC with a linear kernel on the whole dataset and tuning the hyperparametes as explained in the methods section the resulting final model is:

$$SVC(C = 11.630513020006196,$$
$$class\_weight = \{0:1, 1:2\},$$
$$kernel = linear, probability = True)$$

Hyper-parameter distributions

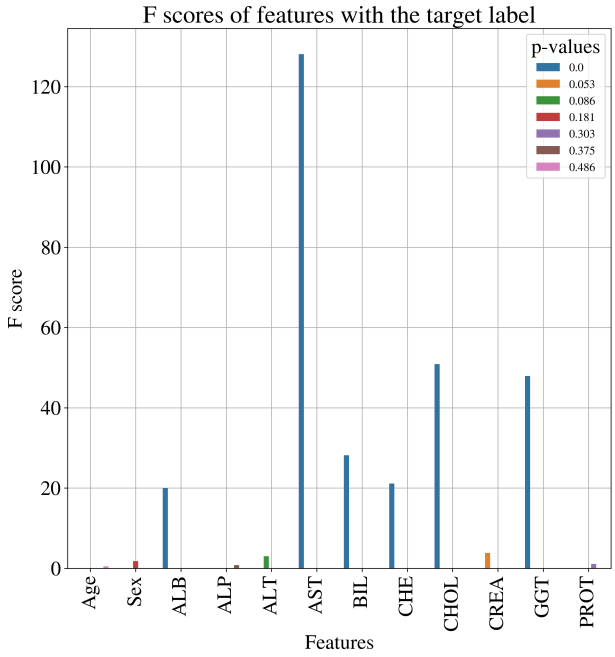
The stored best parameters for each of the 50 cross-validation loops identified by Optuna in the model selection pipeline stage, were utilized to estimate the probability density function of the regularization parameter C for both Support Vector Classifier (SVC) and Logistic Regression (LR) classifiers in the context of HVC prediction. It is worth noting that smaller values of C in both cases result in reduced fixation of the model on the training set, at the cost of increased bias. Therefore, choosing an appropriate value for C is crucial for achieving a balanced trade-off between variance and bias. The distributions of the parameter C are illustrated in (figure 6(a)), for the majority of SVC and LR classifiers the mean of the distributions lies between $10 < mean < 100$ with an original hyper-parameter space of 0.001 to 100.

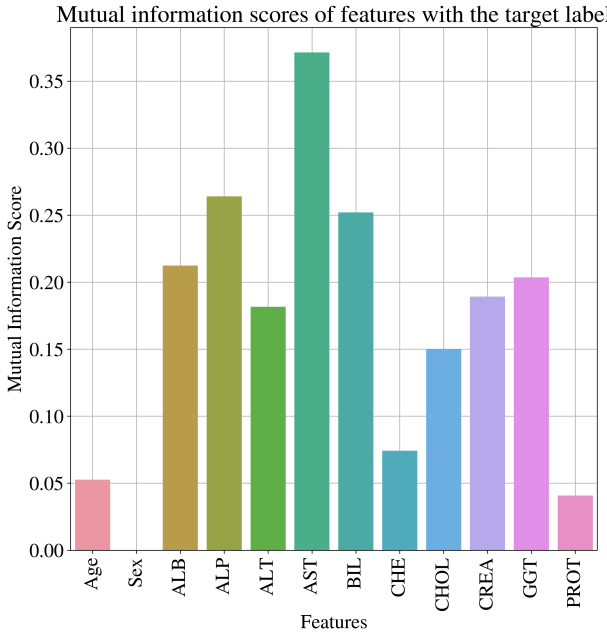
For non-linear SVC classifiers, it is important to consider the hyper-parameter *gamma* in conjunction with the distribution of C. *gamma* controls the shape of the decision boundary and acts as an inverse regularization strength parameter similar to C. We present the joint distributions of $C, gamma$ for non-linear SVC in (figure 6(b)) which showcase clearly their negative covariance.

Results from feature selection

Note: Before presenting the results, it is important to mention that the polynomial kernel SVC has been excluded from this analysis as it failed to converge. This could be attributed to its extensive hyper-parameter space, which made it difficult to reach a decision boundary that accurately separates the two classes with the limited feature space available.

Based on the ANOVA F-score, the top five features in descending order are AST, BIL, CHE, CHOL and GGT. The feature ALB has a slightly lower score than CHE and is therefore not selected. The rest of the features have a score close to zero, as shown in the figure below. On the other hand, the top five features selected based on the Mutual Information (MI) score are ALB, ALP, AST, BIL, GGT, again in descending order. Unlike the ANOVA F-score selection, now with the lack of assumption about specific relationships (e.g. linear), the remaining features such as CHOL, CREA, and ALT seem to carry a significant amount of information about the class label, as presented in the following figure. Running the pipeline with the top five features selected based on mutual information score, we collected the metrics presented in (figure 7(a)). The best algorithm in this feature space is the KNN classifier with a mean MCC score of 0.750, an $\approx$ 19% increase over its mean MCC score without feature selection. The standard deviation of the KNN however is also lower $= 0.85$ compared to the original space $= 0.121$. The only other classifier that benefited in the MI selected feature space is the GNB





classifier with an MCC mean score of 0.743, $\approx$ 3.5% increase with a standard deviation 0.09 lower than the original. Regarding the rest of the classifiers, they performed significantly worse both with lower MCC scores with higher standard deviations. We can assume that KNN performs better due to the increased importance of the notion of Euclidean distance in the lower dimensional feature space. GNB on the other hand may have a better score in this feature space because these features hold better to the Independence assumption than the total of 12 features in the original feature space. Furthermore, the other algorithms lack the necessary information, present in the original feature space to separate the classes effectively. Based on our analysis, we have found that the MI selection-based feature space only yields beneficial results for the KNN and GNB algorithms, but the improved scores for these algorithms are still lower than the performance achieved by the SVC-linear model using the original feature space.

After repeating the pipeline with the top five features selected based on ANOVA F-score, we obtained the metrics presented in Figure 7(b). The best classifier we found was the ridge logistic regression, which achieved a mean MCC score of 0.7914 with a standard deviation of 0.085. Upon examining the classification metrics in more detail , we observed that Precision and Specificity were generally high for all classifiers, the Recall scores were significantly lower, with a higher standard deviation. This suggests that the feature space created by the selected ANOVA features provides more information about the healthy blood donor class than the HCV-positive class. Since Recall is the most critical metric for this task as thoroughly explained before, we believe that feature selection based on ANOVA is not suitable for the dataset.

Conclusions

In the context of HCV prediction, we developed a pipeline to evaluate and select machine learning models for binary classification. Our pipeline includes a custom-built class called *ClassifierCV* that handles imbalanced datasets and weighted recall and precision. We used 50 nested cross-validation loops to assess feature scaling methods and feature selection methods. We found that the linear-SVC model category had the highest mean MCC score of 0.823 with a standard deviation of 0.1. However, reducing the feature space to five features using ANOVA F-score and Mutual Information did not provide a better alternative nor improved the classification performance of the best-selected classifier of the original feature space. Therefore, we recommend considering other feature selection methods such as feature ranking via decision trees and/or increasing the number of top-selected features. Currently, *ClassifierCV* can only evaluate one classifier per class call using nested cross-validation. To improve the pipeline, we suggest allowing *ClassifierCV* to read classifiers from a list internally, eliminating the need for looping and providing a cleaner and more concise pipeline. Regarding imbalanced datasets, the cost-sensitive learning method we used is only one of several methods to deal with them. We recommend

utilizing additional methods, such as oversampling with SMOTE, to extend the base pipeline presented in this report.

References

[1] *Hepatitis C*. en. URL: https://www.who.int/news-room/fact-sheets/detail/hepatitis-c (visited on 04/18/2023).

[2] BR Thapa and Anuj Walia. "Liver function tests and their interpretation". In: *The Indian Journal of Pediatrics* 74 (2007), pp. 663–671.

[3] Takuya Akiba et al. "Optuna: A next-generation hyperparameter optimization framework". In: *Proceedings of the 25th ACM SIGKDD international conference on knowledge discovery & data mining*. 2019, pp. 2623–2631.

[4] Fabian Pedregosa et al. "Scikit-learn: Machine learning in Python". In: *the Journal of machine Learning research* 12 (2011), pp. 2825–2830.

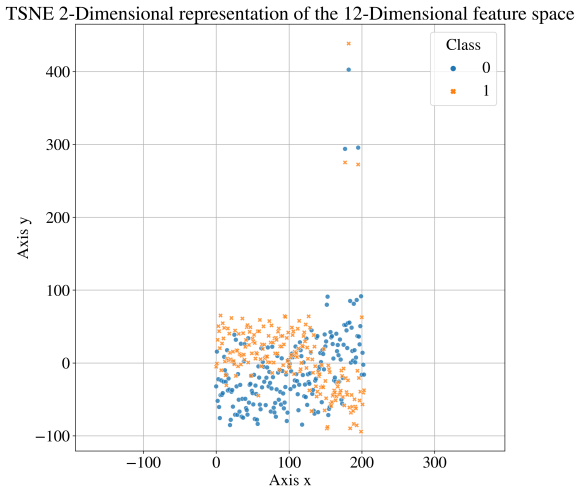
[5] Davide Chicco and Giuseppe Jurman. "The advantages of the Matthews correlation coefficient (MCC) over F1 score and accuracy in binary classification evaluation". In: *BMC genomics* 21 (2020), pp. 1–13.

Figures

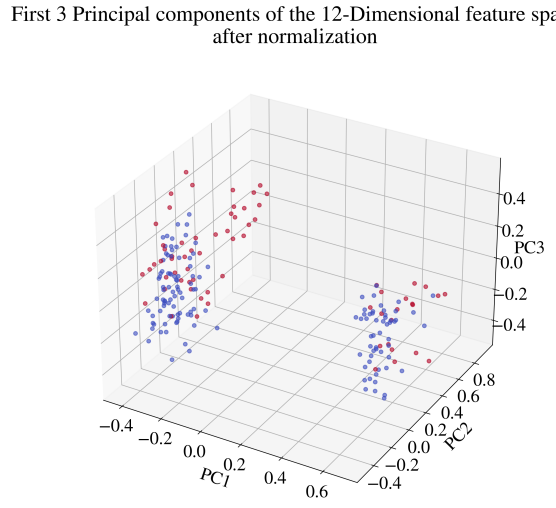
The figures referenced in the methods & results sections of the report are showcased on the following pages: $6 - 10$.

Code Availability

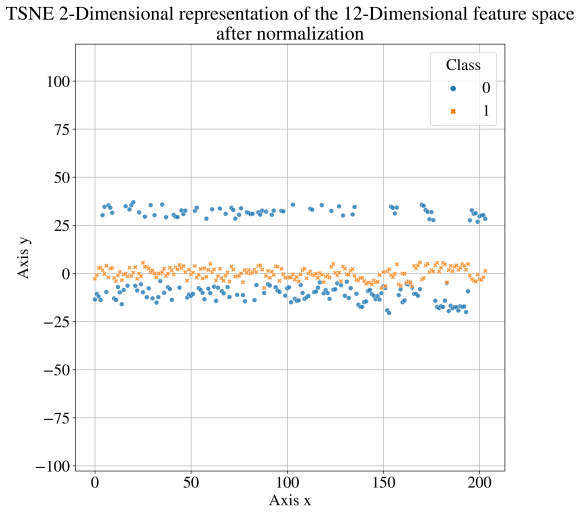
The code to reproduce the results of the paper and/or extend the base pipeline is available in this GitHub repository.



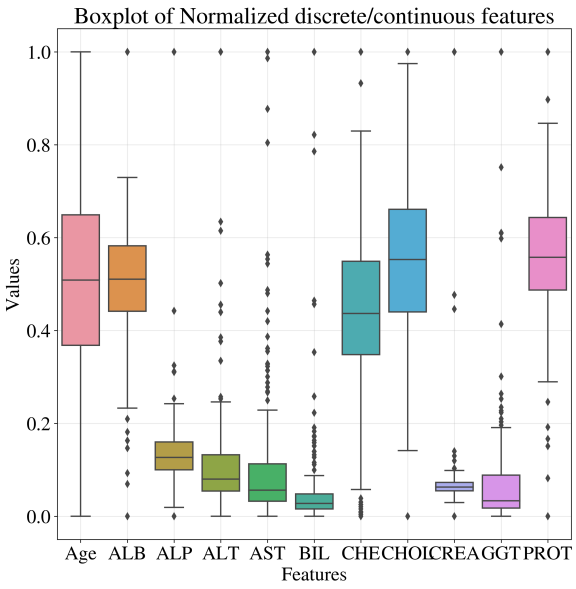
(a) TSNE visualization of the original data-set



(b) First 3 principal components (PCs) of the normalized data-set



(c) TSNE visualization of the normalized data-set



(d) Boxplot of the normalized data-set excluding binary features

Figure 3: *Hepatitis C exploratory data analysis findings*

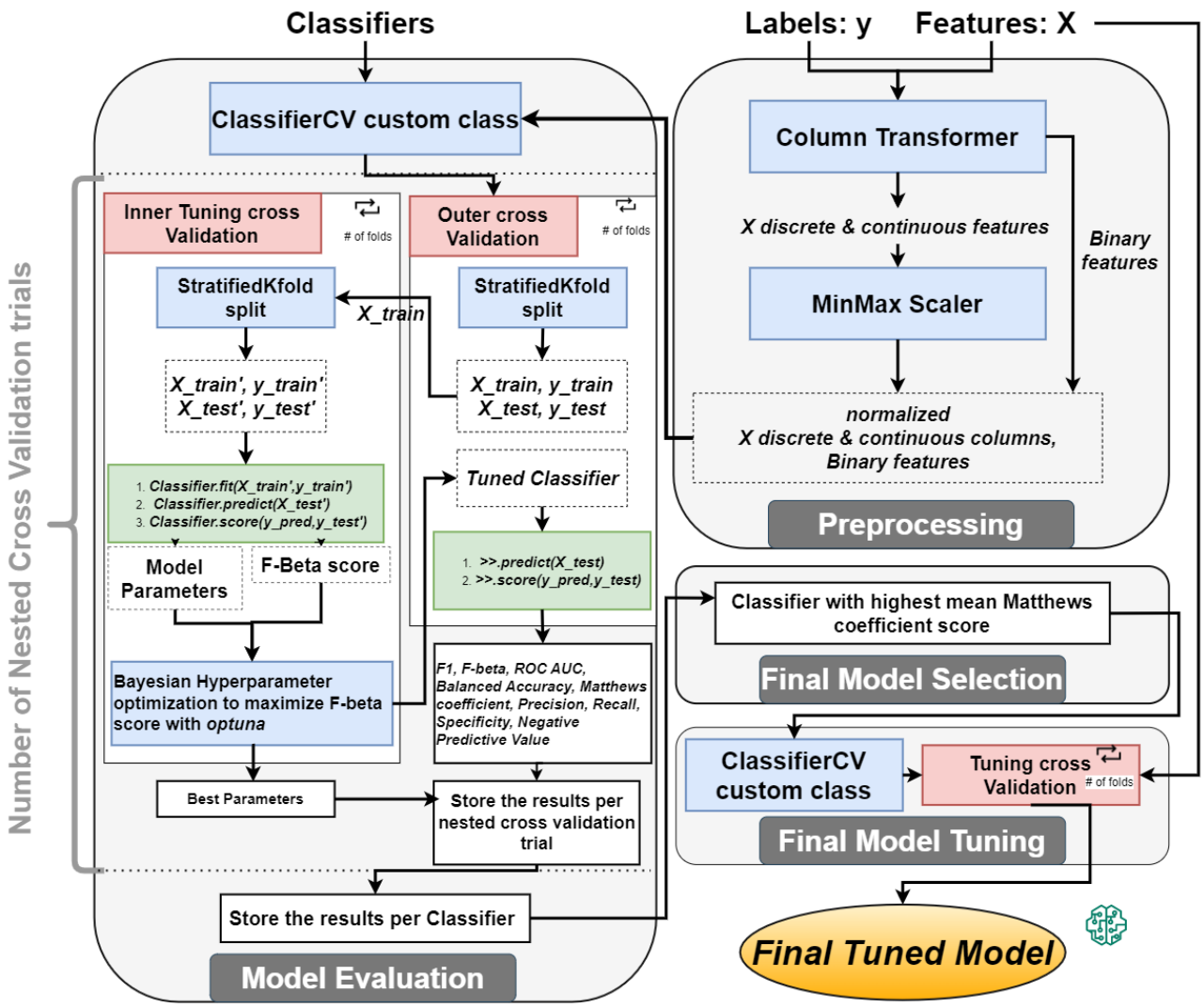
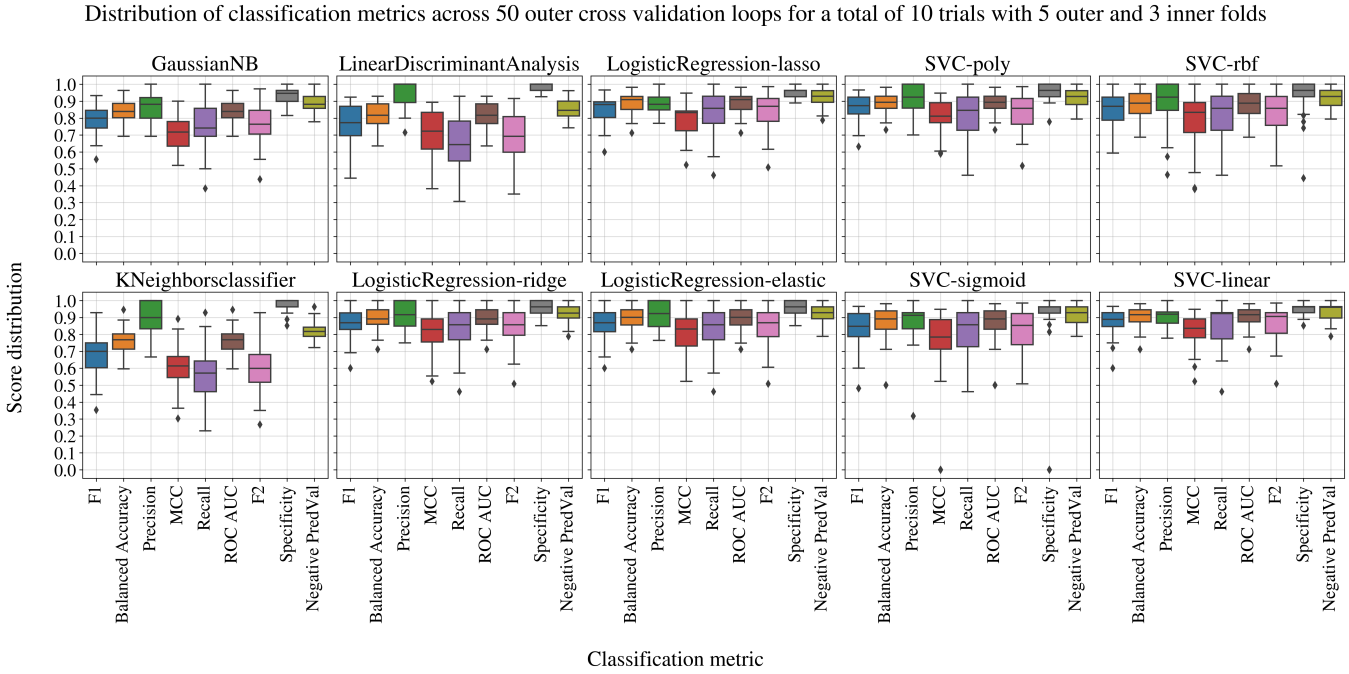
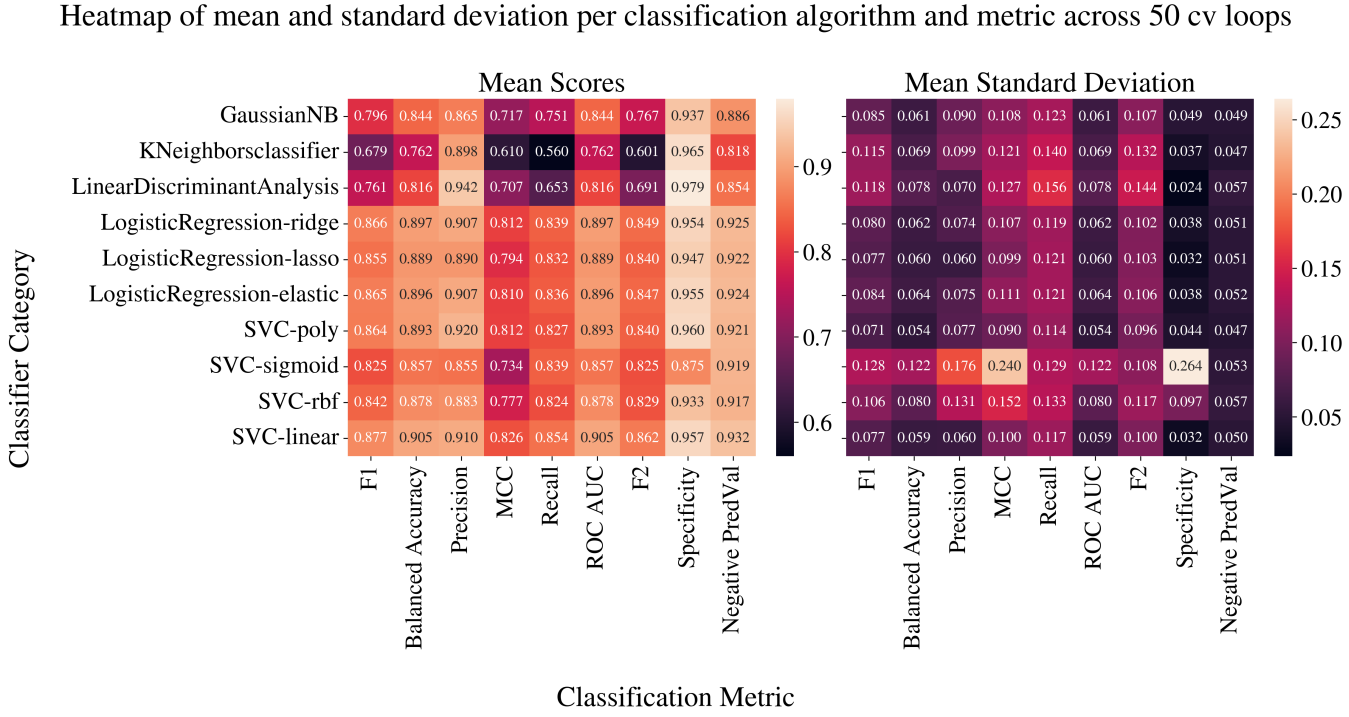


Figure 4: *baseline Hepatitis C machine learning Pipeline*

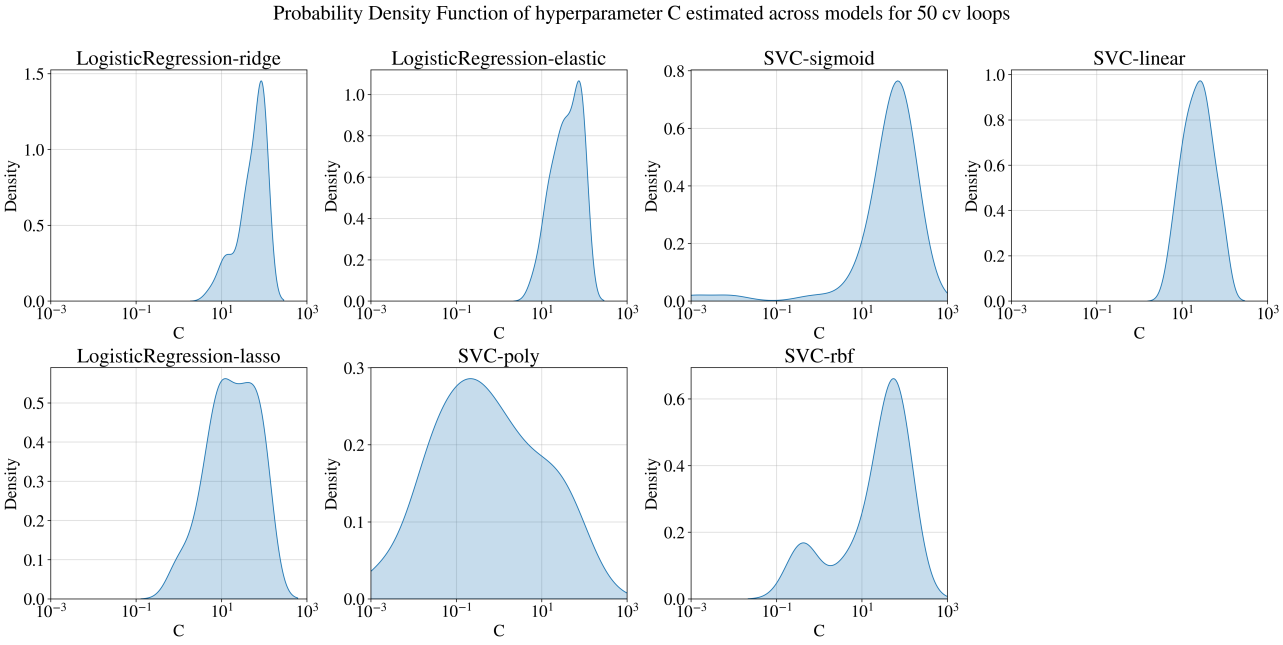


(a) Score distribution for each classifier categories after 50 outer cross-validation loops

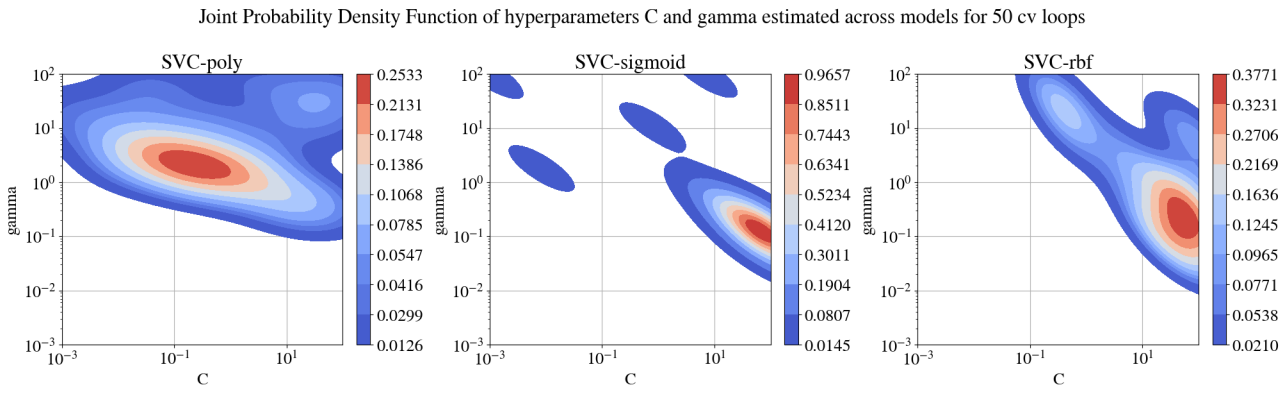


(b) Exact score mean and standard deviation for each classifier categories after 50 outer cross-validation loops

Figure 5: *Results from the model evaluation pipeline stage*

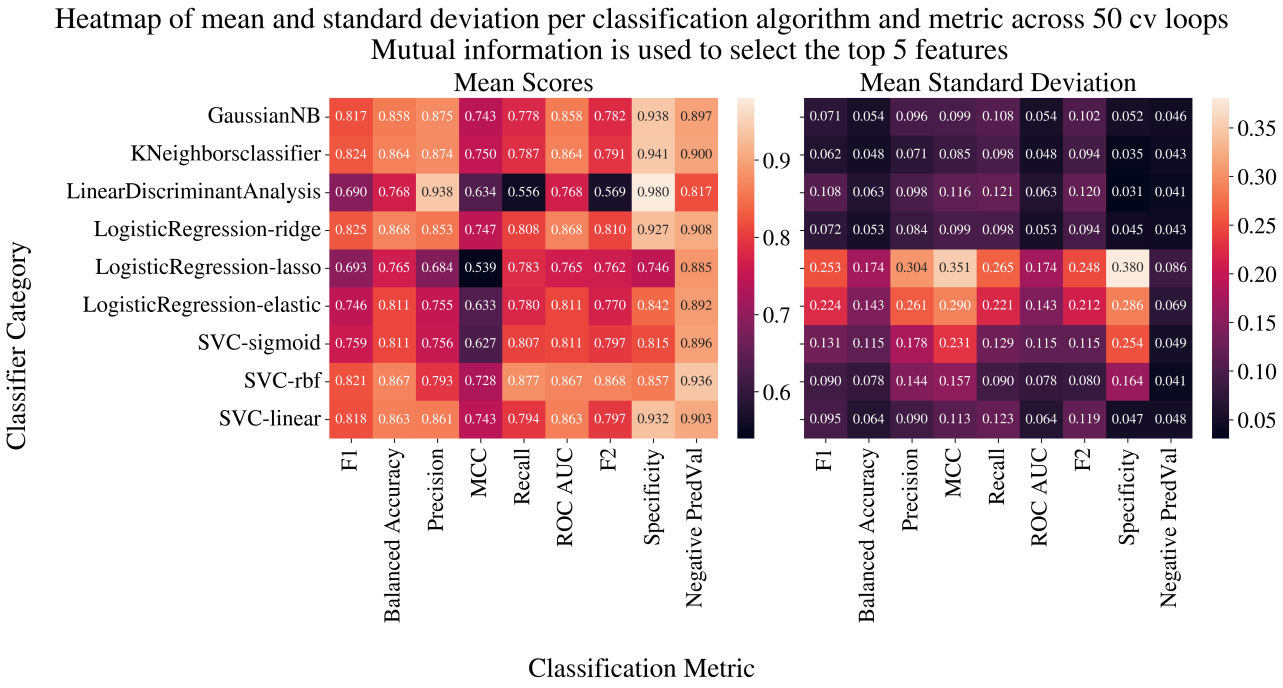


(a) Hyper-parameter C distribution for all applicable classifiers

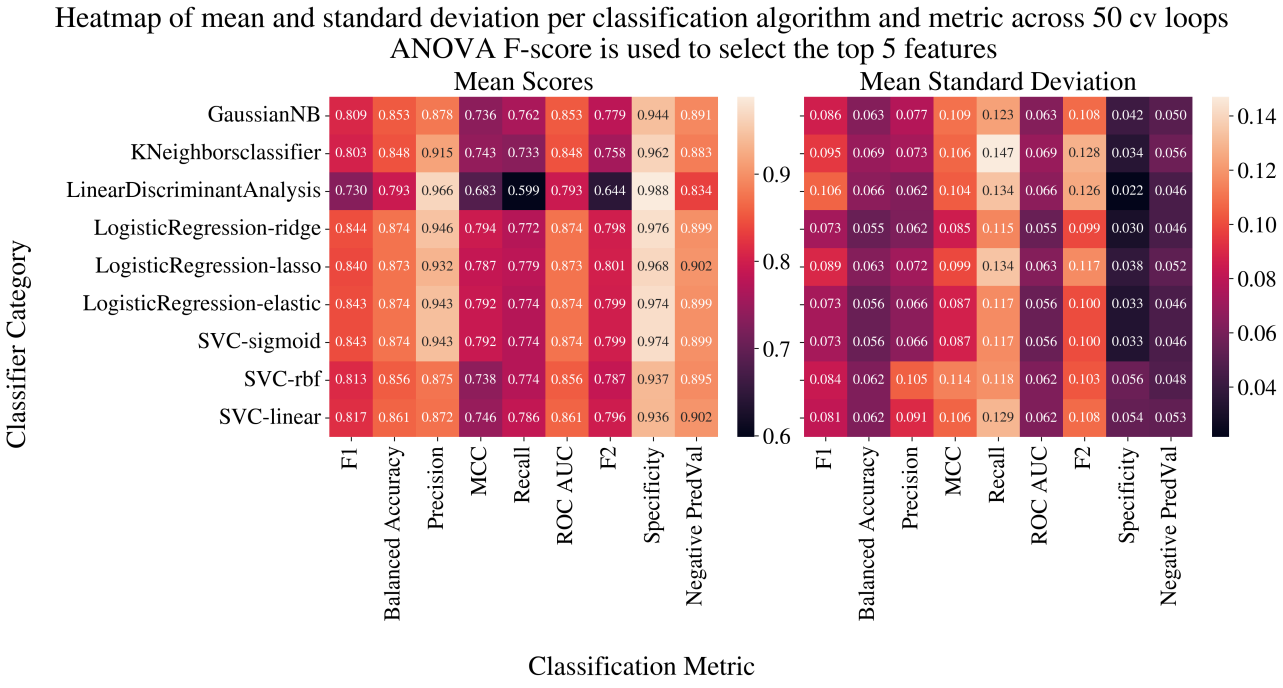


(b) Hyper-parameter C, gamma joint distribution for all applicable classifiers

Figure 6: *Hyper-parameter distribution per classifier algorithm across the 50 cv loops*



(a) Classification score based on the MI selected features



(b) Classification score based on the ANOVA F-score selected features

Figure 7: *Results from the feature selection performance*