Ovarian Cancer Histotypes: Report of Statistical Findings

Derek Chiu

June 18, 2025

Table of contents

List of Figures

List of Tables

Preface

This report of statistical findings describes the classification of ovarian cancer histotypes using data from NanoString CodeSets.

Marina Pavanello conducted the initial exploratory data analysis, Cathy Tang implemented class imbalance techniques, Derek Chiu conducted the normalization and statistical analysis, and Lauren Tindale and Aline Talhouk are the project leads.

1 Introduction

Ovarian cancer has five major histotypes: high-grade serous carcinoma (HGSC), low-grade serous carcinoma (LGSC), endometrioid carcinoma (ENOC), mucinous carcinoma (MUC), and clear cell carcinoma (CCOC). A common problem with classifying these histotypes is that there is a class imbalance issue. HGSC dominates the distribution, commonly accounting for 70% of cases in many patient cohorts, while the other four histotypes are spread over the rest of the cases. Subsampling methods like up-sampling, down-sampling, and SMOTE can be used to mitigate this problem.

The supervised learning is performed under a consensus framework: we consider various classification algorithms and use evaluation metrics like accuracy, F1-score, and Kappa, to inform the decision of which methods to carry forward for prediction in confirmation and validation sets.

2 Methods

2.1 Pre-Processing

2.1.1 Case Selection

Prior to pre-processing, samples were split into a training, a confirmation, and a validation set.

- Training
  - CS1: OOU, OOUE, VOA, MAYO, MTL
  - CS2: OOU, OOUE, VOA, MAYO, OVAR3, OVAR11, JAPAN, MTL, POOL-CTRL
  - CS3: OOU, OOUE, VOA, POOL-1, POOL-2, POOL-3
- Confirmation:
  - CS3: TNCO
- Validation:
  - CS3: DOVE4

2.1.2 Quality Control

Before normalization, we calculated several quality control measures and excluded samples that failed to achieve sample quality in one or more of these measures.

- **Linearity of positive control genes**: If the R-squared from a linear model of positive controls and their concentrations is less than 0.95 or missing, then the sample is flagged.
- **Imaging quality**: The sample is flagged if the field of view percentage is less than 75%.
- **Positive Control flag**: We consider the two smallest positive controls at concentrations 0.5 and 1. If these two controls are less than the lower limit of detection (defined as two standard deviations below the mean of the negative control expression), or if the mean negative control expression is 0, the sample is flagged.
- **The signal-to-noise ratio or percent of genes detected**: These two measures are defined as the ratio of the average housekeeping gene expression over the upper limit of detection, defined as two standard deviations above the mean of the negative control expression (or 0 if this limit is less than 0.001), and the proportion of endogenous genes with expression greater than the upper limit of detection. These measures are flagged if they are below a pre-specified threshold, which is determined visually by considering their bivariate distribution in a scatterplot. In this case, we used 100 for the SNR threshold and 50% for the threshold for genes detected. Note: these thresholds were determined by examining the relationship in Section 3.3.2.

2.1.3 Housekeeping Genes Normalization

The full training set (n=1257) comprised of data from three CodeSets (CS) 1, 2, and 3. Data normalization removes technical variation from high-throughput platforms to improve the validity of comparative analyses.

Each CodeSet was first normalized to housekeeping genes: *ACTB*, *RPL19*, *POLR1B*, *SDHA*, and *PGK1*. Housekeeping genes encode proteins responsible for basic cell function and have consistent expression in all cells. All expression values were log2 transformed. Normalization to housekeeping genes corrects the viable RNA from each sample. This is achieved by subtracting the average log (base 2)-transformed expression of the housekeeping genes from the log (base 2)-transformed expression of each gene:

$$log_2(\text{endogenous gene expression}) - \text{average}(log_2(\text{housekeeping gene expresssion})) = \text{relative expression} \tag{2.1}$$

2.1.4 Between CodeSet and Site Normalization

To normalize between CodeSets, we randomly selected five specimens, one from each histotype, among specimens repeated in all three CodeSets. This formed the reference set (Random 1). We selected only one sample from each histotype to use as few samples as possible for normalization and retain the rest for analysis.

A reference-based approach (Talhouk et al. (2016)) was used to normalize CS1 to CS3 and CS2 to CS3 across their common genes:

$$\text{X-Norm}_{\text{CS1}} = X_{\text{CS1}} + \bar{R}_{\text{CS3}} - \bar{R}_{\text{CS1}}\text{X-Norm}_{\text{CS2}} = X_{\text{CS2}} + \bar{R}_{\text{CS3}} - \bar{R}_{\text{CS2}} \tag{2.2}$$

Samples in CS3 were processed at three different locations; we also had to normalize for "site" in this CodeSet. Finally, the CS3 expression samples were included in the training set without further normalization:

$$\text{X-Norm}_{\text{CS3-USC}} = X_{\text{CS3-USC}} + \bar{R}_{\text{CS3-VAN}} - \bar{R}_{\text{CS3-USC}}\text{X-Norm}_{\text{CS3-AOC}} = X_{\text{CS3-AOC}} + \bar{R}_{\text{CS3-VAN}} - \bar{R}_{\text{CS3-AOC}} \tag{2.3}$$

Finally, the CS3 expression samples were included in the training set without further normalization. The initial training set is assembled by combining all four of the previously mentioned normalized datasets along with the two CS3 expression subsets not used in normalization:

$$\begin{aligned}\text{Training Set} &= \text{X-Norm}_{\text{CS1}} + \text{X-Norm}_{\text{CS2}} + \text{X-Norm}_{\text{CS3-USC}} + \text{X-Norm}_{\text{CS3-AOC}} + \text{X-Norm}_{\text{CS3}} + \text{X-Norm}_{\text{CS3-VAN}} \\ &= \text{X-Norm}_{\text{CS1}} + \text{X-Norm}_{\text{CS2}} + \text{X-Norm}_{\text{CS3}}\end{aligned} \tag{2.4}$$

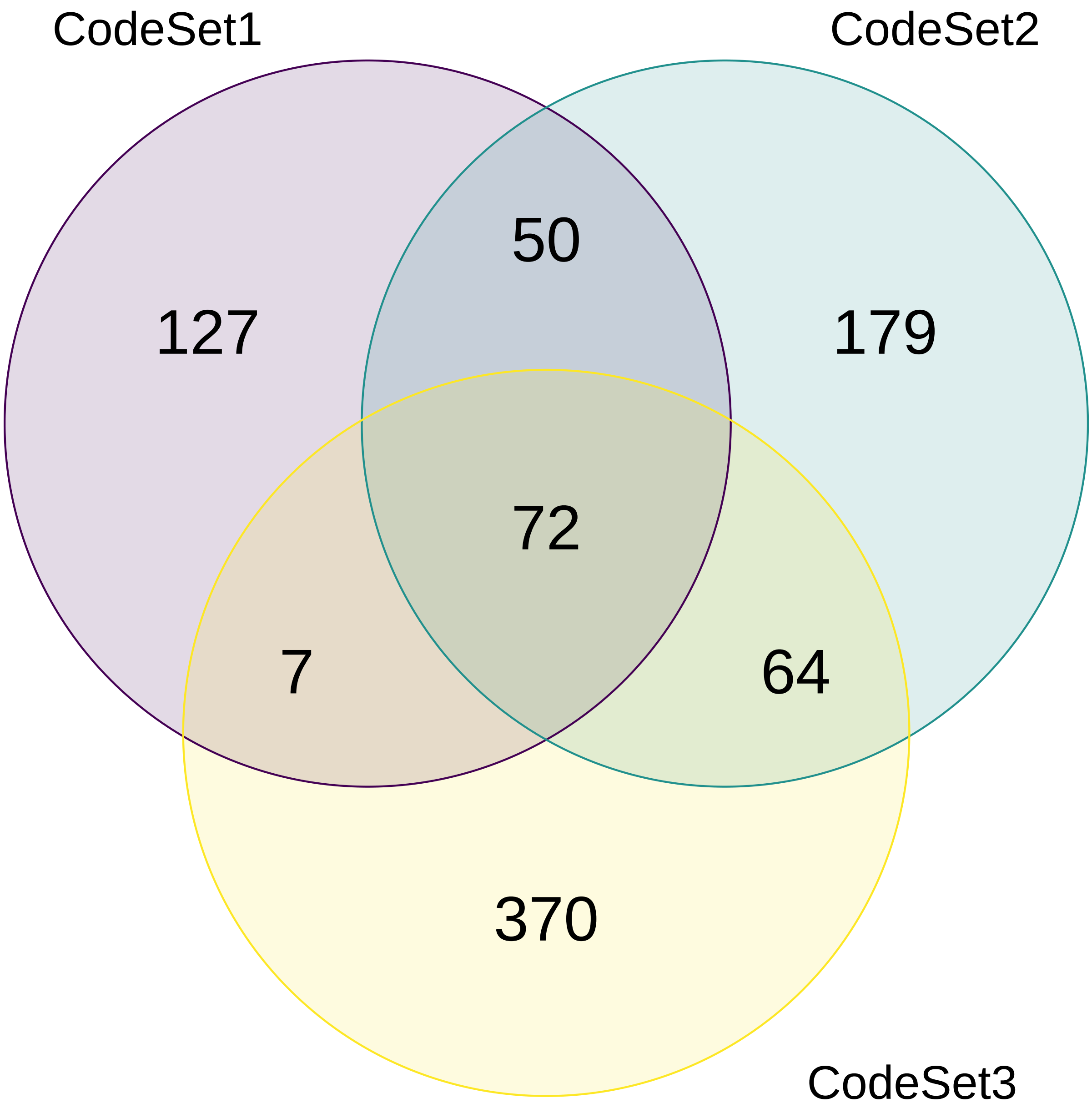

Figure 2.1: Venn diagram of common and unique gene targets covered by each CodeSet

2.1.5 Final Processing

We map ovarian histotypes to all remaining samples and keep the major histotypes for building the predictive model: high-grade serous carcinoma (HGSC), clear cell ovarian carcinoma (CCOC), endometrioid ovarian carcinoma (ENOC), low-grade serous carcinoma (LGSC), mucinous carcinoma (MUC).

Duplicate cases (two samples with the same ottaID) were removed before generating the final training set to use for fitting the classification models. All CS3 cases were preferred over CS1 and CS2, and CS3-Vancouver cases were preferred over CS3-AOC and CS3-USC when selecting duplicates.

The final training set used only genes that were common across all three CodeSets.

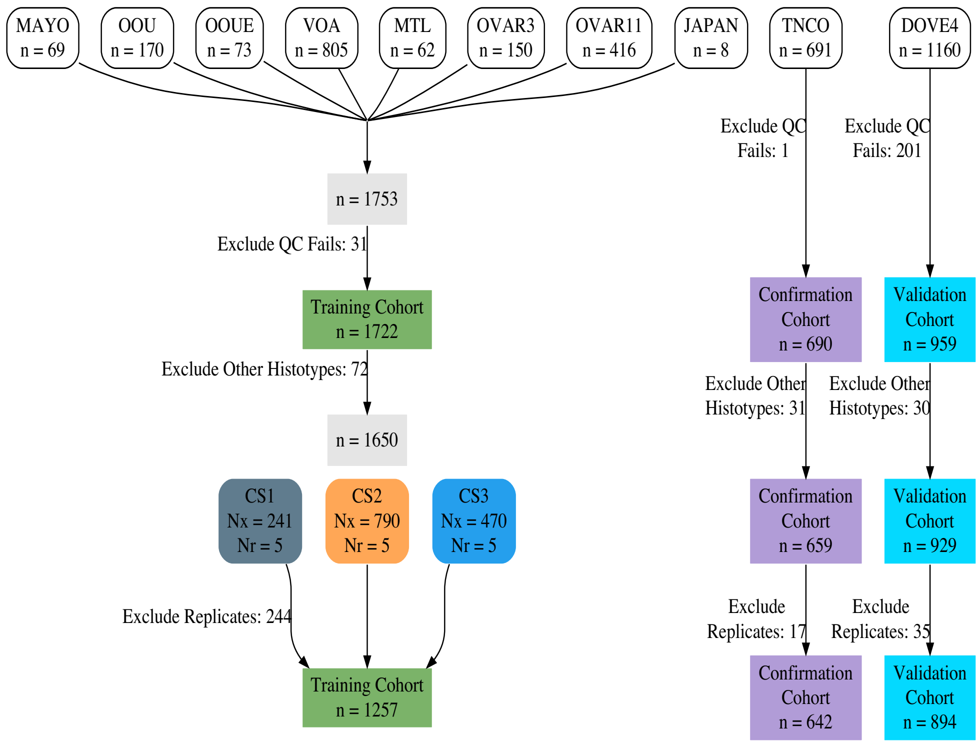

Figure 2.2: Cohorts Selection

2.2 Classifiers

We use 4 classification algorithms in the supervised learning framework for the Training Set. The pipeline was run using SLURM batch jobs submitted to a partition on a CentOS 7 server. All resampling techniques, pre-processing, model specification, hyperparameter tuning, and evaluation metrics were implemented using the `tidymodels` suite of packages. The classifiers we used are:

- Random Forest (`rf`)
- Support Vector Machine (`svm`)
- XGBoost (`xgb`)
- Regularized Multinomial Regression (`mr`)

2.2.1 Resampling of Training Set

We used a nested cross-validation design to assess each classifier while also performing hyperparameter tuning. An outer 5-fold CV stratified by histotype was used together with an inner 5-fold CV with 2 repeats stratified by histotype. This design was chosen such that the test sets of the inner resamples would still have a reasonable number of samples belonging to the smallest minority class.

The outer resampling method cannot be the bootstrap, because the inner training and inner test sets will likely contain the same samples as a result of sampling with replacement in the outer training set. This phenomenon might result in inflated performance as some observations are used both to train and evaluate the hyperparameter tuning in the inner loop.

2.2.2 Hyperparameter Tuning

The following specifications for each classifier were used for tuning hyperparameters:

- `rf` and `xgb`: The number of trees were fixed at 500. Other hyperparameters were tuned across 10 randomly selected points in a latin hypercube design.
- `svm`: Both the cost and sigma hyperparameters were tuned across 10 randomly selected points in a latin hypercube design. We tuned the cost parameter in the range [1, 8]. The range for tuning the sigma parameter was obtained from the 10% and 90% quantiles of the estimation using the `kernlab::sigest()` function.
- `mr`: We generated 10 randomly selected points in a latin hypercube design for the penalty (lambda) parameter. Then, we generated 10 evenly spaced points in [0, 1] for the mixture (alpha) parameter in the regularized multinomial regression model. These two sets of 10 points were crossed to generate a tuning grid of 100 points.

The hyperparameter combination that resulted in the highest average F1-score across the inner training sets was selected for each classifier to use as the model for assessing prediction performance in the outer training loop.

2.2.3 Subsampling

Here are the specifications of the subsampling methods used to handle class imbalance:

- None: No subsampling is performed
- Down-sampling: All levels except the minority class are sampled down to the same frequency as the minority class
- Up-sampling: All levels except the majority class are sampled up to the same frequency as the majority class
- SMOTE: All levels except the majority class have synthetic data generated until they have the same frequency as the majority class
- Hybrid: All levels except the majority class have synthetic data generated up to 50% of the frequency of the majority class, then the majority class is sampled down to the same frequency as the rest.

The figure below helps visualize how the distribution of classes changes when we apply subsampling techniques to handle class imbalance:

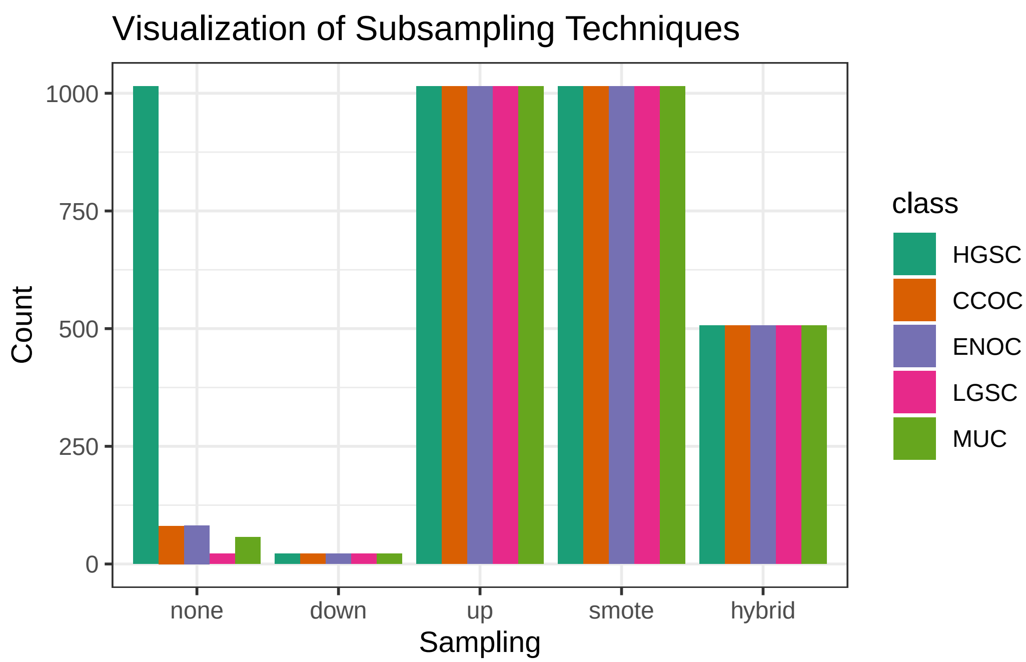

Figure 2.3: Visualization of Subsampling Techniques

2.2.4 Workflows

The 4 **algorithms** and 5 **subsampling** methods are crossed to create 20 different classification **workflows**. For example, the `hybrid_xgb` workflow is a classifier that first pre-processes a training set by applying a hybrid subsampling method, and then proceeds to use the XGBoost algorithm to classify ovarian histotypes.

2.3 Two-Step Algorithm

The HGSC histotype comprises of approximately 80% of cases among ovarian carcinoma patients, while the remaining 20% of cases are relatively, evenly distributed among ENOC, CCOC, LGSC, and MUC histotypes. We can implement a two-step algorithm as such:

- Step 1: use binary classification for HGSC vs. non-HGSC
- Step 2: use multinomial classification for the remaining non-HGSC classes

Let

$$
\begin{aligned}
X_k &= \text{Training data with k classes} \\
C_k &= \text{Class with highest } F_1 \text{ score from training } X_k \\
W_k &= \text{Workflow associated with } C_k
\end{aligned}
\tag{2.5}
$$

Figure 2.4 shows how the two-step algorithm works:

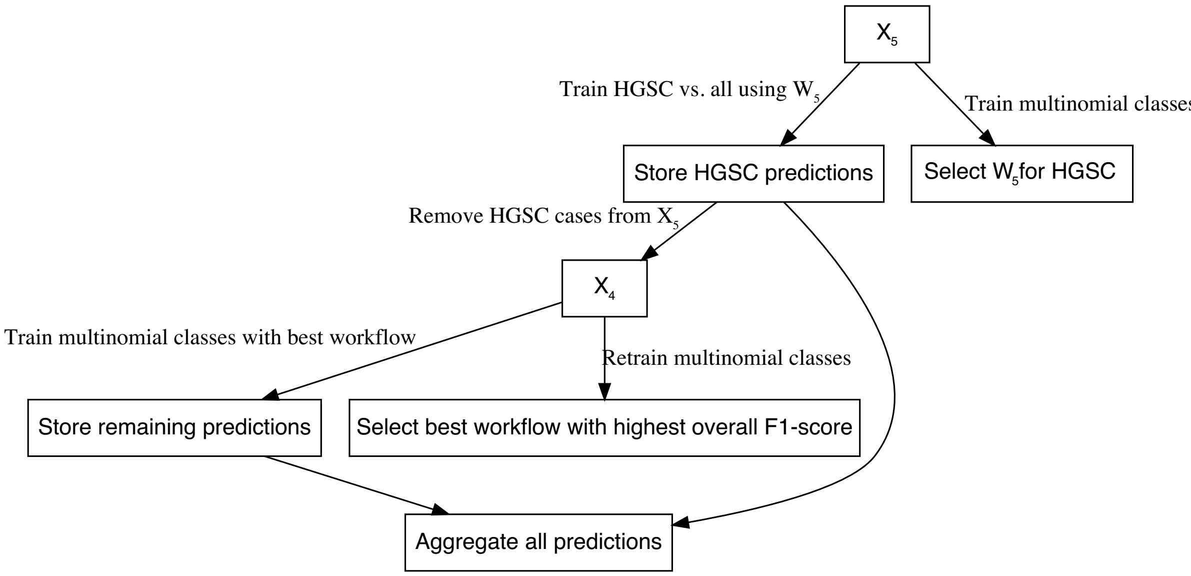

Figure 2.4: Two-Step Algorithm

2.3.1 Aggregating Predictions

The aggregation for two-step predictions is quite straightforward:

1. Predict HGSC vs. non-HGSC
2. Among all non-HGSC cases, predict CCOC vs. LGSC vs. MUC vs. ENOC

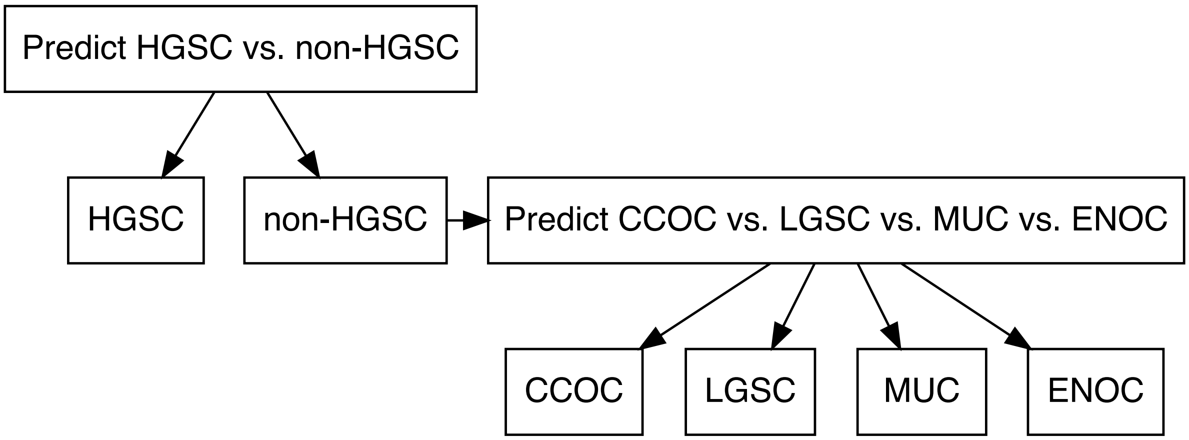

Figure 2.5: Aggregating Predictions for Two-Step Algorithm

2.4 Sequential Algorithm

Instead of training on k classes simultaneously using multinomial classifiers, we can use a sequential algorithm that performs k-1 one-vs-all binary classifications iteratively to obtain a final prediction of all cases. At each step in the sequence, we classify one class vs. all other classes, where the classes that make up the “other” class are those not equal to the current “one” class and excluding all “one” classes from previous steps. For example, if the “one” class in step 1 was HGSC, the “other” classes would include CCOC, ENOC, LGSC, and MUC. If the “one” class in step 2 was CCOC, the “other” classes include ENOC, LGSC, and MUC.

The order of classes and workflows to use at each step in the sequential algorithm must be determined using a retraining procedure. After removing the data associated with a particular class, we retrain using the remaining data using multinomial classifiers as described before. The class and workflow to use for the next step in the sequence is selected based on the best per-class evaluation metric value (e.g. F1-score).

Figure 2.6 illustrates how the sequential algorithm works for K=5, using ovarian histotypes as an example for the classes.

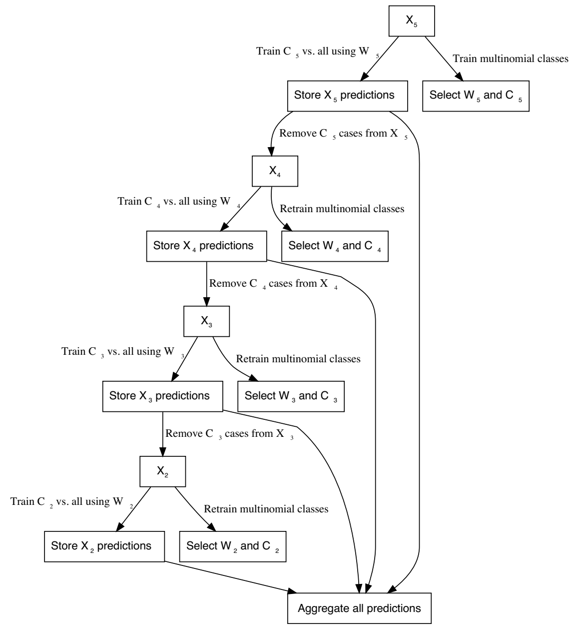

Figure 2.6: Sequential Algorithm

The subsampling method used in the first step of the sequential algorithm is used in all subsequent steps in order to maintain data pre-processing consistency. As a result, we are only comparing classification algorithms within one subsampling method across the entire sequential algorithm.

2.4.1 Aggregating Predictions

We have to aggregate the one-vs-all predictions from each of the sequential algorithm workflows in order to obtain a final class prediction on a holdout test set. Each sequential workflow has to be assessed on every sample to ensure that cases classified into the "all" class from a previous step of the sequence are eventually assigned a predicted class. For example, say that based on certain class-specific metrics we determined that the order of classes in the sequential algorithm was to predict HGSC vs. non-HGSC, CCOC vs. non-CCOC, LGSC vs. non-LGSC, and then MUC vs. ENOC. Figure 2.7 illustrates how the final predictions are assigned:

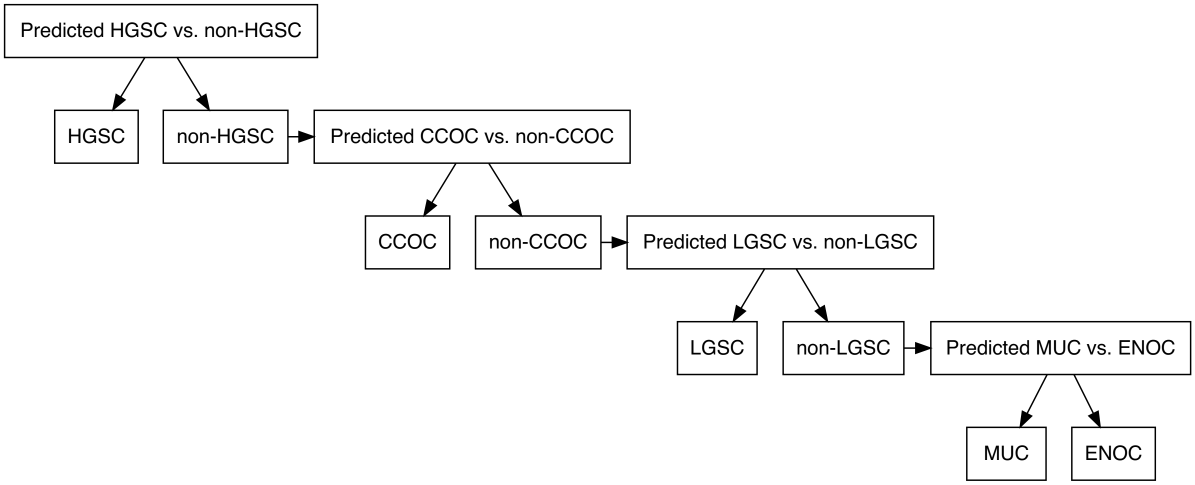

Figure 2.7: Aggregating Predictions for Sequential Algorithm

2.5 Performance Evaluation

2.5.1 Class Metrics

We use the accuracy, sensitivity, specificity, F1-score, kappa, balanced accuracy, and geometric mean, as class metrics to measure both training and test performance between different workflows. Multiclass extensions of these metrics can be calculated except for F1-score, where we use macro-averaging to obtain an overall metric. Class-specific metrics are calculated by recoding classes into one-vs-all categories for each class.

2.5.1.1 Accuracy

The accuracy is defined as the proportion of correct predictions out of all cases:

$$\text{accuracy} = \frac{TP}{TP + FP + FN + TN} \tag{2.6}$$

2.5.1.2 Sensitivity

Sensitivity is the proportional of correctly predicted positive cases, out of all cases that were truly positive

$$\text{sensitivity} = \frac{TP}{TP + FN} \tag{2.7}$$

2.5.1.3 Specificity

Specificity is the proportional of correctly predicted negative cases, out of all cases that were truly negative.

$$\text{specificity} = \frac{TN}{TN + FP} \tag{2.8}$$

2.5.1.4 F1-Score

The F-measure can be thought of as a harmonic mean between precision and recall:

$$F_{meas} = \frac{(1 + \beta^2) \times precision \times recall}{(\beta^2 \times precision) + recall} \tag{2.9}$$

The β value can be adjusted to place more weight upon precision or recall. The most common value is β is 1, which is also commonly known as the F1-score. A multiclass extension doesn't exist for the F1-score, so we use macro-averaging to calculate this metric when there are more than two classes. For example, with k classes, the macro-averaged F1-score is equal to:

$$F_{1_{macro}} = \frac{1}{k} \sum_{i=1}^{k} F_{1_i} \tag{2.10}$$

where each F_{1_i} is the F1-score computed frrom recoding classes into $k = i$ vs. $k \neq i$.

In situations where there is not at least one predicted case for each of the classes (e.g. for a poor classifier), F_{1_i} is undefined because the per-class precision of class i is undefined. Those F_{1_i} terms are removed from the $F_{1_{macro}}$ equation and the resulting value may be inflated. Interpreting the F1-score in such a case would be misleading.

2.5.1.5 Balanced Accuracy

Balanced accuracy is the arithmetic mean of sensitivity and specificity.

$$\text{Balanced Accuracy} = \frac{\text{Sensitivity} + \text{Specificity}}{2} \tag{2.11}$$

2.5.1.6 Kappa

Kappa is the defined as:

$$\text{kappa} = \frac{p_0 - p_e}{1 - p_e} \tag{2.12}$$

where p_0 is the observed agreement among raters and p_e is the hypothetical probability of agreement due to random chance.

2.5.2 AUC

The area under the receiver operating curve (AUC) is calculated by adding up the area under the curve formed by plotting sensitivity vs. 1 - specificity. The Hand-till method is used as a multiclass extension for the AUC.

We did not use AUC to measure class-specific training set performance because combining predicted probabilities in a one-vs-all fashion might be potentially misleading. The sum of probabilities that add up to the "other" class is not equivalent to the predicted probability of the "other" class when using a multiclass classifier.

Instead, we only reported ROC curves and their associated AUCs for test set performance among the highest ranked algorithms.

2.6 Rank Aggregation

To select the best algorithm, we implemented a two-stage rank aggregation procedure using the Genetic Algorithm. First, we ranked all workflows based on per-class F1-scores, balanced accuracy, and kappa to see which workflows performed well in predicting all five histotypes. Then, we took the ranks from these three performance metrics and performed a second run of rank aggregation. The top 5 workflows were determined from the final rank aggregation result.

2.7 Gene Optimization

We want to discover an optimal set of genes for the classifiers while including specific genes from other studies such as PrOTYPE and SPOT. A total of 72 genes are used in the classifier training set.

There are 16 genes in the classifier set that overlap with the PrOTYPE classifier: COL11A1, CD74, CD2, TIMP3, LUM, CYTIP, COL3A1, THBS2, TCF7L1, HMGA2, FN1, POSTN, COL1A2, COL5A2, PDZK1IP1, FBN1.

There are also 13 genes in the classifier set that overlap with the SPOT signature: HIF1A, CXCL10, DUSP4, SOX17, MITF, CDKN3, BRCA2, CEACAM5, ANXA4, SERPINE1, TCF7L1, CRABP2, DNAJC9.

We obtain a total of 28 genes from the union of PrOTYPE and SPOT genes that we want to include in the final classifier, regardless of model performance. We then incrementally add genes one at a time from the remaining 44 candidate genes based on a variable importance rank to the set of 28 base genes and recalculate performance metrics. The number of genes at which the performance peaks or starts to plateau may indicate an optimal gene set model for us to compare with the full set model.

Here is the breakdown of genes used and whether they belong to the PrOTYPE and/or SPOT sets:

Table 2.1: Gene Distribution

Genes	PrOTYPE	SPOT
TCF7L1	v	v
COL11A1	v	
CD74	v	
CD2	v	
TIMP3	v	
LUM	v	
CYTIP	v	
COL3A1	v	
THBS2	v	
HMGA2	v	
FN1	v	
POSTN	v	
COL1A2	v	
COL5A2	v	
PDZK1IP1	v	
FBN1	v	
HIF1A		v
CXCL10		v
DUSP4		v
SOX17		v
MITF		v
CDKN3		v
BRCA2		v
CEACAM5		v
ANXA4		v
SERPINE1		v
CRABP2		v
DNAJC9		v
C10orf116		
GAD1		
TPX2		
KGFLP2		
EGFL6		
KLK7		
PBX1		

LIN28B
TFF3
MUC5B
FUT3
STC1
BCL2
PAX8
GCNT3
GPR64
ADCYAP1R1
IGKC
BRCA1
IGJ
TFF1
MET
CYP2C18
CYP4B1
SLC3A1
EPAS1
HNF1B
IL6
ATP5G3
DKK4
SENP8
CAPN2
C1orf173
CPNE8
IGFBP1
WT1
TP53
SEMA6A
SERPINA5
ZBED1
TSPAN8
SCGB1D2
LGALS4
MAP1LC3A

2.7.1 Variable Importance

Variable importance is calculated using either a model-based approach if it is available, or a permutation-based VI score otherwise. The variable importance scores are averaged across the outer training folds, and then ranked from highest to lowest.

For the sequential and two-step classifiers, we calculate an overall VI rank by taking the cumulative union of genes at each variable importance rank across all sequences, until all genes have been included.

The variable importance measures are:

- Random Forest: impurity measure (Gini index)
- XGBoost: gain (fractional contribution of each feature to the model based on the total gain of the corresponding features's splits)
- SVM: permutation based p-values
- Multinomial regression: absolute value of estimated coefficients at cross-validated lambda value

3 Distributions

3.1 Histotype Distribution

Table 3.1: Histotype Distribution in Training Set by Processing Stage

Variable	Levels	CS1	CS2	CS3	Total
Selected Cohorts					
Histotype	HGSC	128 (44%)	655 (73%)	1808 (73%)	2591 (71%)
	CCOC	48 (16%)	62 (7%)	164 (7%)	274 (7%)
	ENOC	60 (20%)	49 (5%)	250 (10%)	359 (10%)
	MUC	17 (6%)	58 (6%)	68 (3%)	143 (4%)
	LGSC	19 (6%)	20 (2%)	36 (1%)	75 (2%)
	Other	22 (7%)	59 (7%)	151 (6%)	232 (6%)
Total	N (%)	294 (8%)	903 (25%)	2477 (67%)	3674 (100%)
QC					
Histotype	HGSC	122 (43%)	641 (73%)	1676 (74%)	2439 (71%)
	CCOC	48 (17%)	62 (7%)	158 (7%)	268 (8%)
	ENOC	60 (21%)	47 (5%)	213 (9%)	320 (9%)
	MUC	16 (6%)	56 (6%)	65 (3%)	137 (4%)
	LGSC	18 (6%)	20 (2%)	36 (2%)	74 (2%)
	Other	22 (8%)	56 (6%)	125 (5%)	203 (6%)
Total	N (%)	286 (8%)	882 (26%)	2273 (66%)	3441 (100%)
Main Histotypes					
Histotype	HGSC	122 (46%)	641 (78%)	1676 (78%)	2439 (75%)
	CCOC	48 (18%)	62 (8%)	158 (7%)	268 (8%)
	ENOC	60 (23%)	47 (6%)	213 (10%)	320 (10%)
	MUC	16 (6%)	56 (7%)	65 (3%)	137 (4%)
	LGSC	18 (7%)	20 (2%)	36 (2%)	74 (2%)
Total	N (%)	264 (8%)	826 (26%)	2148 (66%)	3238 (100%)
Removed Duplicates					
	HGSC	118 (48%)	623 (78%)	1578 (78%)	2319 (76%)

Histotype	CCOC	45 (18%)	56 (7%)	146 (7%)	247 (8%)
	ENOC	56 (23%)	43 (5%)	200 (10%)	299 (10%)
	MUC	13 (5%)	54 (7%)	55 (3%)	122 (4%)
	LGSC	14 (6%)	19 (2%)	32 (2%)	65 (2%)
Total	N (%)	246 (8%)	795 (26%)	2011 (66%)	3052 (100%)
Normalized and Recombined					
Histotype	HGSC	117 (49%)	622 (79%)	454 (97%)	1193 (79%)
	CCOC	44 (18%)	55 (7%)	4 (1%)	103 (7%)
	ENOC	55 (23%)	42 (5%)	4 (1%)	101 (7%)
	MUC	12 (5%)	53 (7%)	4 (1%)	69 (5%)
	LGSC	13 (5%)	18 (2%)	4 (1%)	35 (2%)
Total	N (%)	241 (16%)	790 (53%)	470 (31%)	1501 (100%)
Removed Replicates					
Histotype	HGSC	9 (12%)	552 (78%)	454 (97%)	1015 (81%)
	ENOC	38 (49%)	40 (6%)	4 (1%)	82 (7%)
	CCOC	24 (31%)	53 (7%)	4 (1%)	81 (6%)
	MUC	3 (4%)	50 (7%)	4 (1%)	57 (5%)
	LGSC	3 (4%)	15 (2%)	4 (1%)	22 (2%)
Total	N (%)	77 (6%)	710 (56%)	470 (37%)	1257 (100%)

Table 3.2: Histotype Distribution in Training, Confirmation, and Validation Sets

Variable	Levels	Training	Confirmation	Validation
Histotype	HGSC	1015 (81%)	424 (66%)	699 (78%)
	CCOC	81 (6%)	72 (11%)	69 (8%)
	ENOC	82 (7%)	107 (17%)	88 (10%)
	MUC	57 (5%)	27 (4%)	23 (3%)
	LGSC	22 (2%)	12 (2%)	15 (2%)
Total	N (%)	1257 (45%)	642 (23%)	894 (32%)

3.2 Cohort Distribution

Table 3.3: Pre-QC Cohort Distribution by CodeSet

CodeSet	CS1 N = 294	CS2 N = 903	CS3 N = 2,477
Cohort			
OOU	108 (37%)	43 (4.8%)	19 (0.8%)
OOUE	32 (11%)	30 (3.3%)	11 (0.4%)
VOA	145 (49%)	122 (14%)	538 (22%)
OVAR3	0 (0%)	150 (17%)	0 (0%)
OVAR11	0 (0%)	416 (46%)	0 (0%)
MAYO	6 (2.0%)	63 (7.0%)	0 (0%)
DOVE4	0 (0%)	0 (0%)	1,160 (47%)
TNCO	0 (0%)	0 (0%)	691 (28%)
MTL	3 (1.0%)	59 (6.5%)	0 (0%)
JAPAN	0 (0%)	8 (0.9%)	0 (0%)
POOL-CTRL	0 (0%)	12 (1.3%)	0 (0%)
POOL-1	0 (0%)	0 (0%)	31 (1.3%)
POOL-2	0 (0%)	0 (0%)	14 (0.6%)
POOL-3	0 (0%)	0 (0%)	13 (0.5%)

[1] n (%)

3.3 Quality Control

3.3.1 Failed Samples

We use an aggregated `QCFlag` that considers a sample to have failed QC if any of the following QC conditions are flagged:

- Linearity
- Imaging
- Smallest Positive Control
- Normality

Table 3.4: Quality Control Summary

Quality Control Flag	CS1 N = 294	CS2 N = 903	CS3 N = 2,477
Linearity			
Failed	0 (0%)	4 (0.4%)	0 (0%)
Passed	294 (100%)	899 (100%)	2,477 (100%)
Imaging			
Failed	3 (1.0%)	0 (0%)	4 (0.2%)
Passed	291 (99%)	903 (100%)	2,473 (100%)
Smallest Positive Control			
Failed	0 (0%)	2 (0.2%)	0 (0%)
Passed	294 (100%)	901 (100%)	2,477 (100%)
Normality			
Failed	5 (1.7%)	19 (2.1%)	200 (8.1%)
Passed	289 (98%)	884 (98%)	2,277 (92%)
Overall QC			
Failed	8 (2.7%)	21 (2.3%)	204 (8.2%)
Passed	286 (97%)	882 (98%)	2,273 (92%)

[1] n (%)

3.3.2 %GD vs. SNR

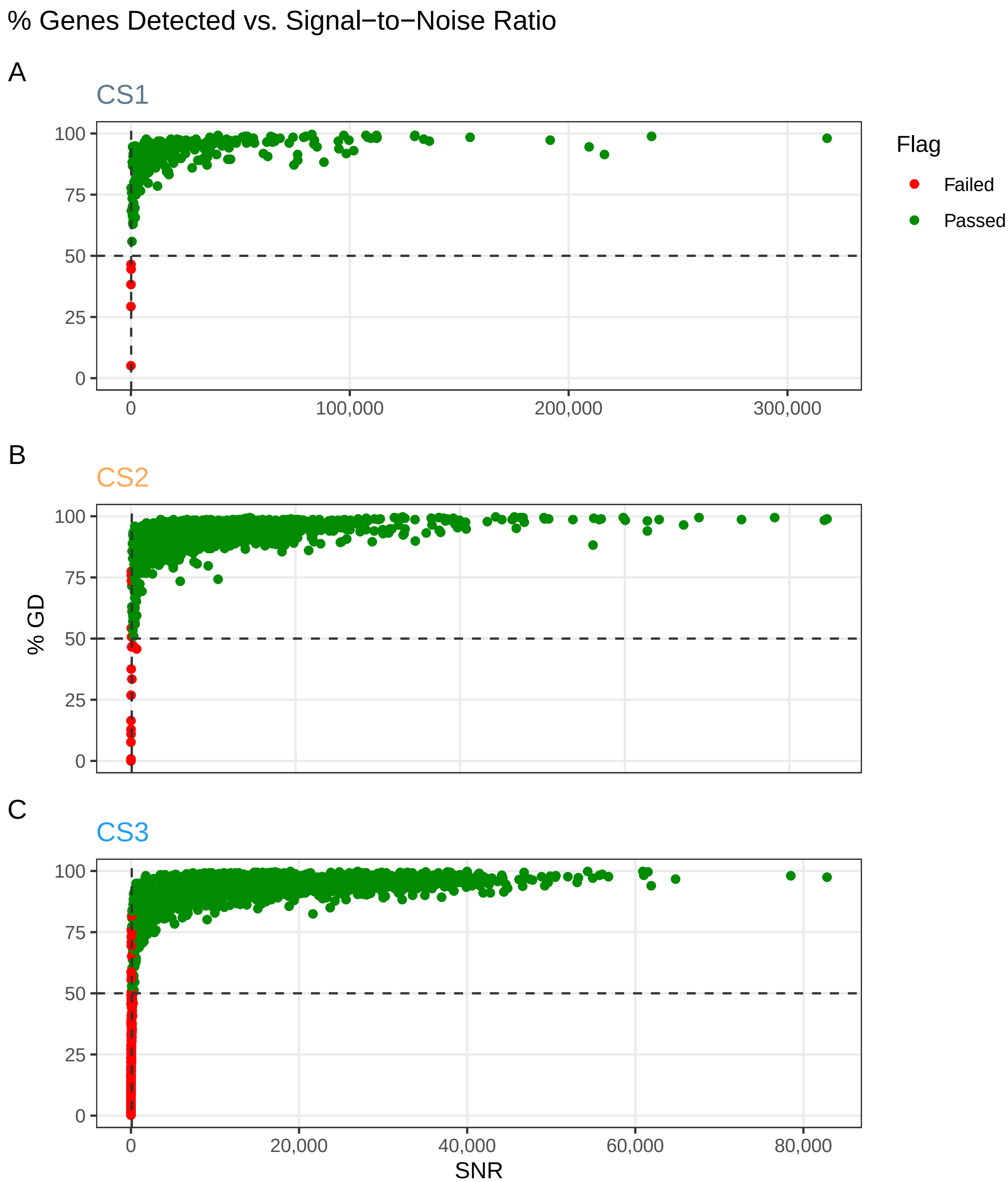

Figure 3.1: % Genes Detected vs. Signal to Noise Ratio

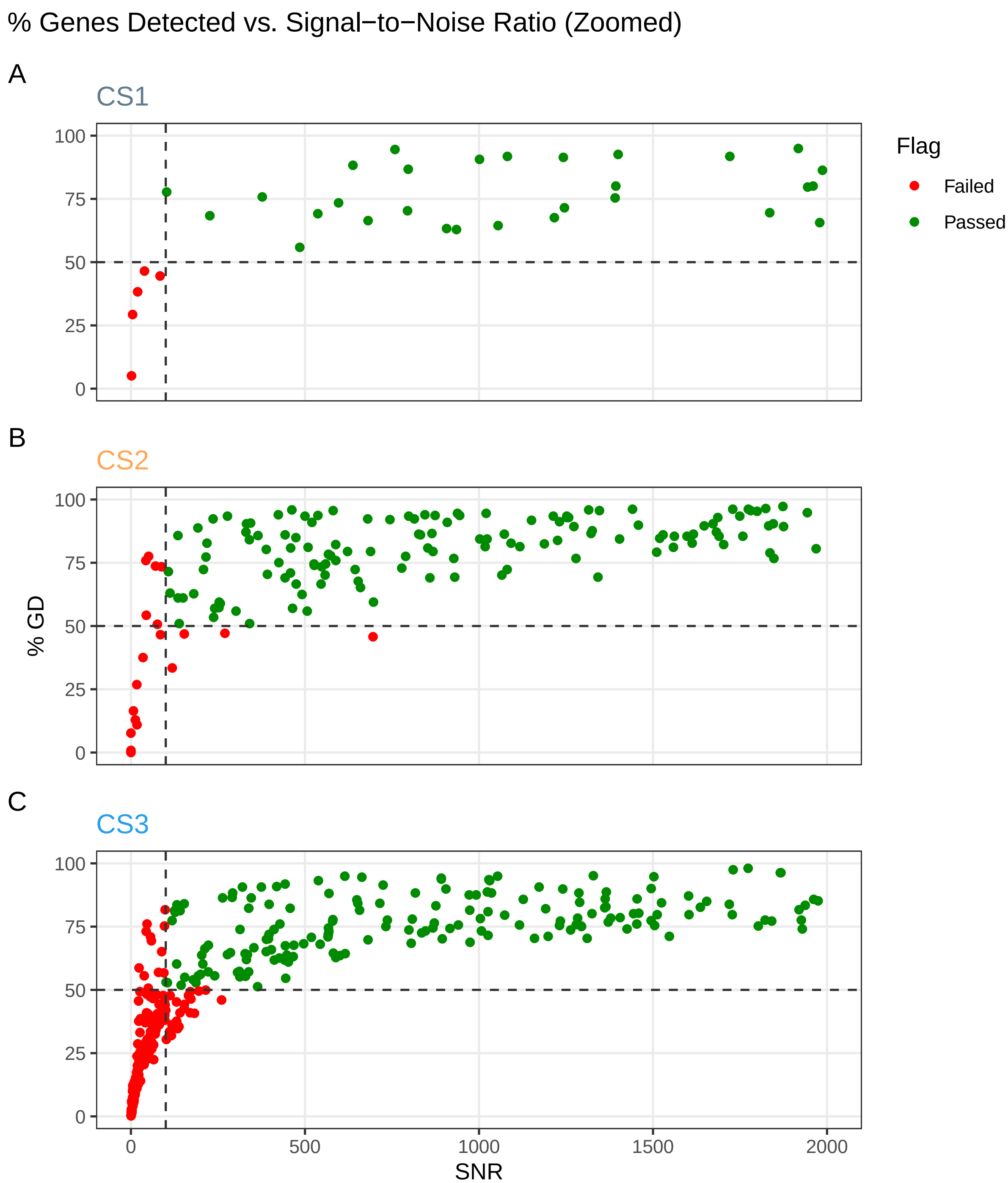

Figure 3.2: % Genes Detected vs. Signal to Noise Ratio (Zoomed)

3.4 Pairwise Gene Expression

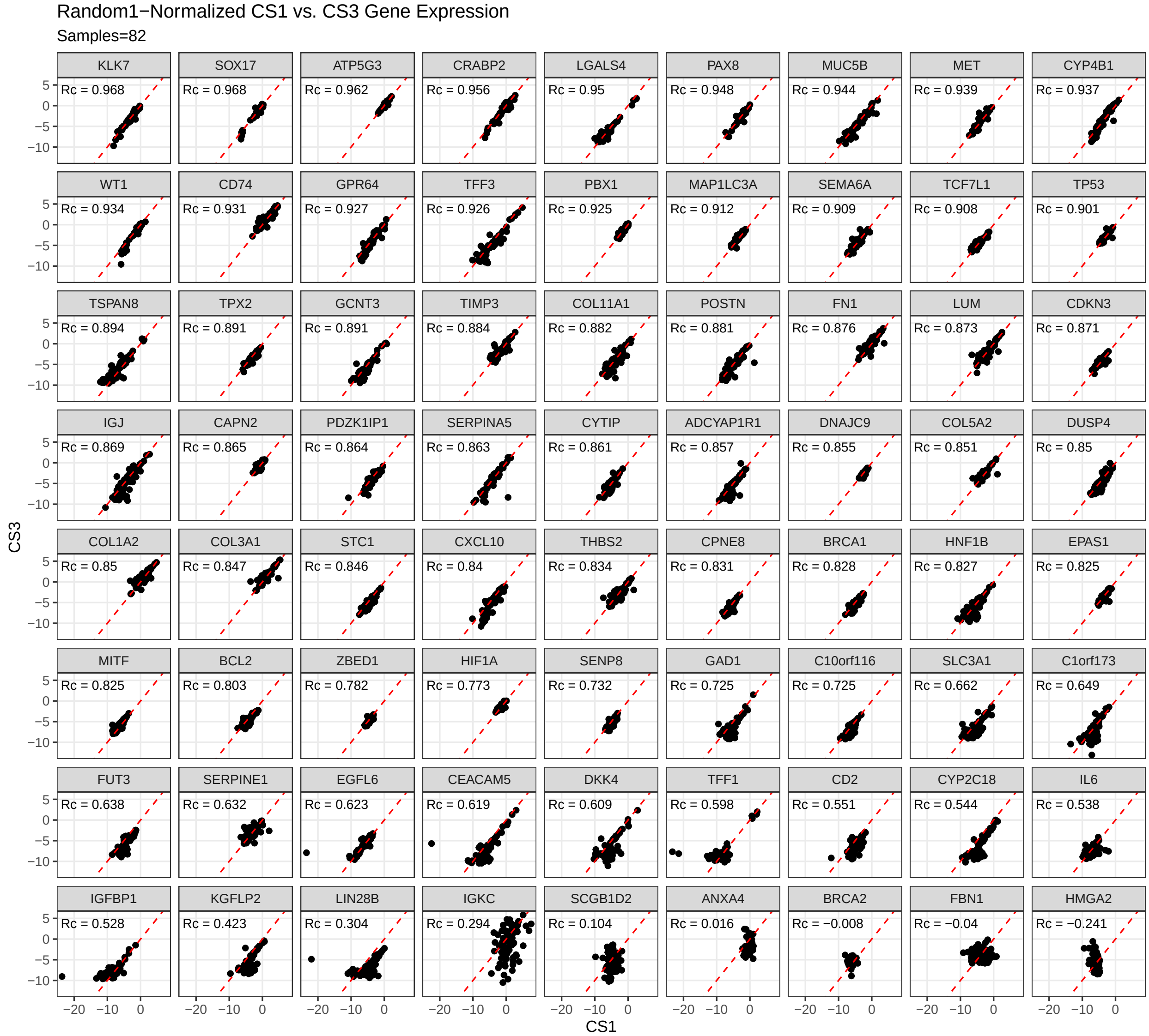

Figure 3.3: Random1-Normalized CS1 vs. CS3 Gene Expression

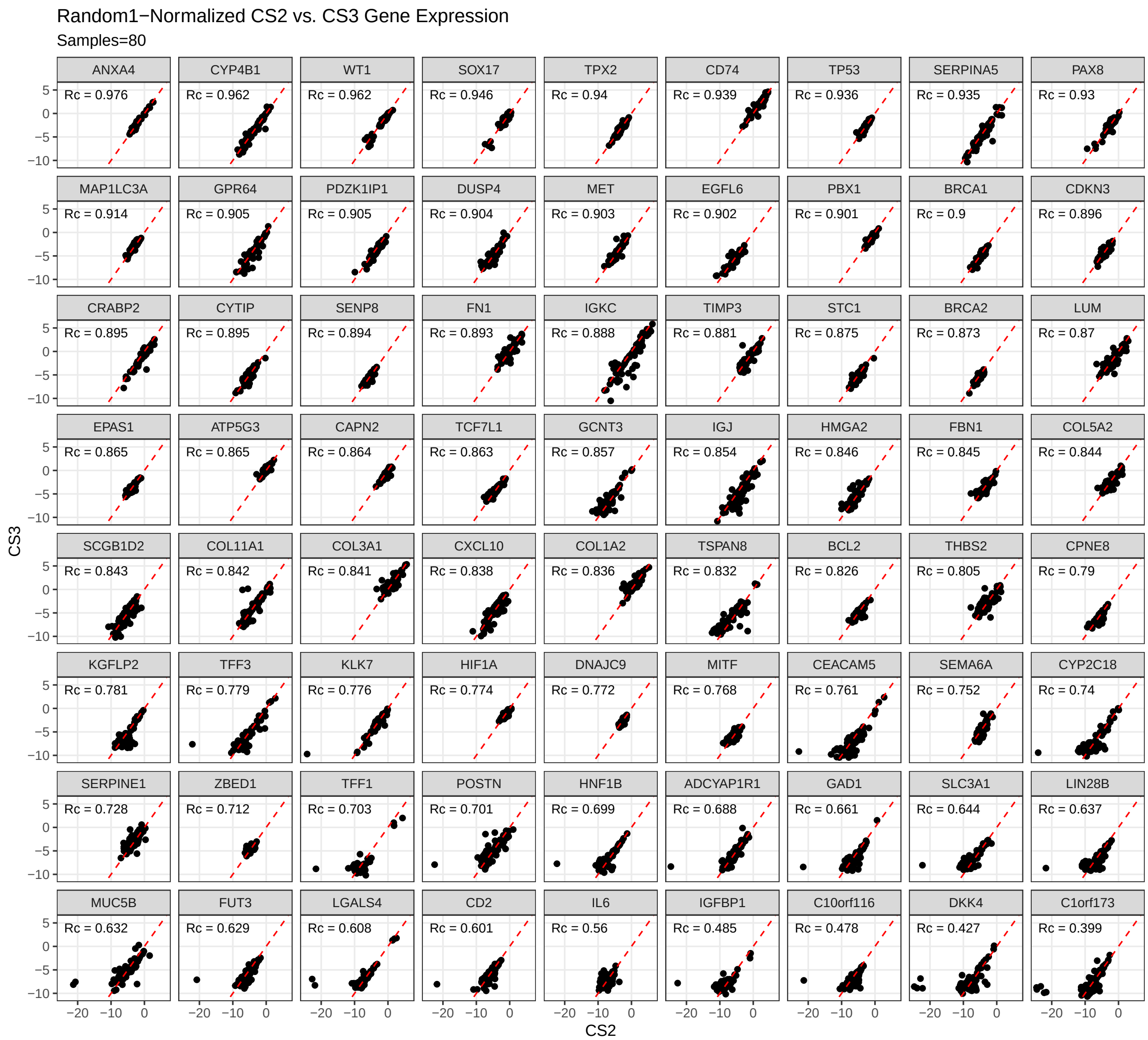

Figure 3.4: Random1-Normalized CS2 vs. CS3 Gene Expression

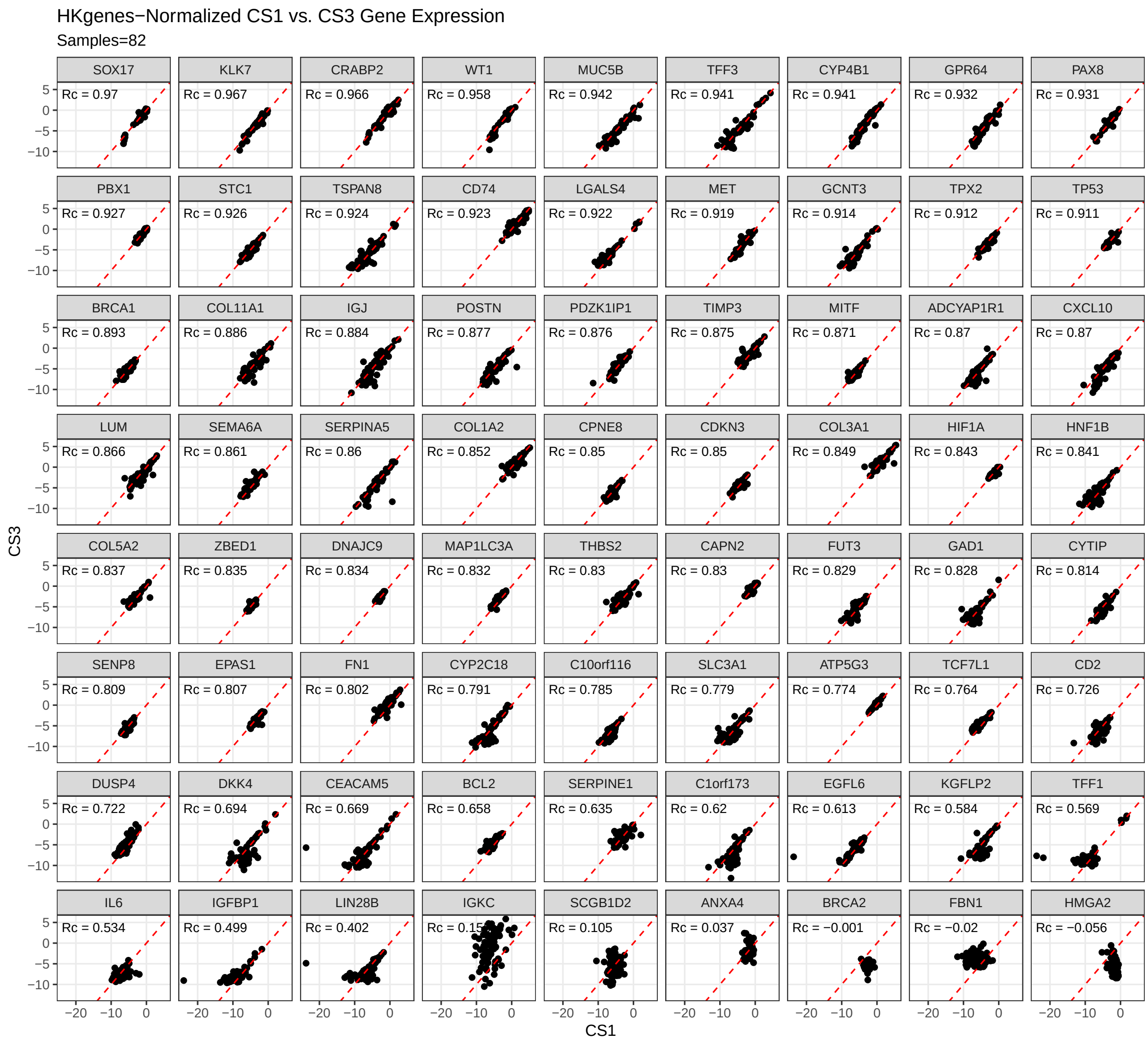

Figure 3.5: HKgenes-Normalized CS1 vs. CS3 Gene Expression

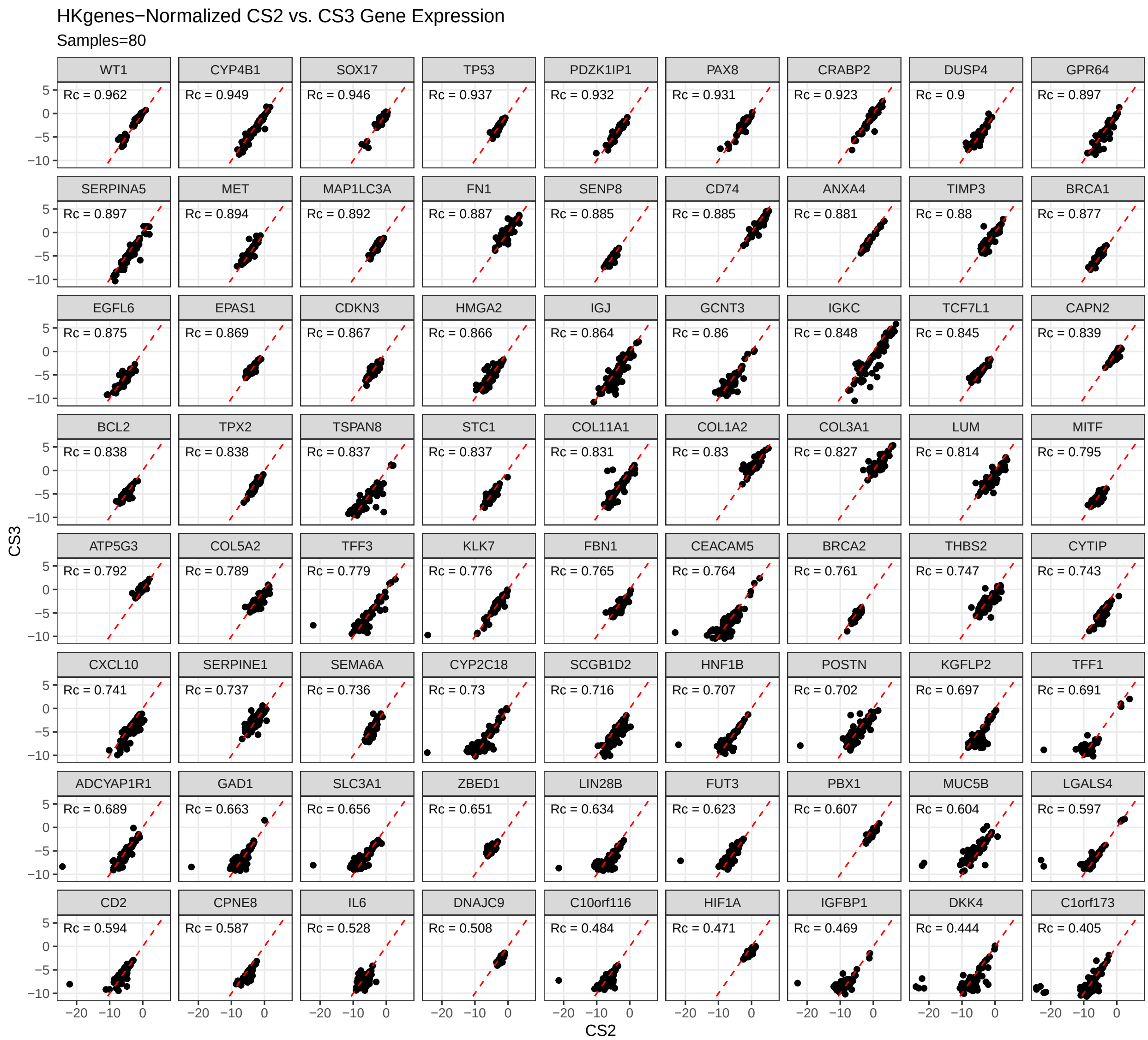

Figure 3.6: HKgenes-Normalized CS2 vs. CS3 Gene Expression

4 Results

We summarize cross-validated training performance of class metrics in the training set. The accuracy, F1-score, and kappa, are the metrics of interest. Workflows are ordered by their mean estimates across the outer folds of the nested CV for each metric.

4.1 Training Set

4.1.1 Accuracy

Table 4.1: Training Set Mean Accuracy

			Histotypes				
Subsampling	Algorithms	Overall	HGSC	CCOC	ENOC	LGSC	MUC
none	rf	0.912	0.935	0.982	0.949	0.982	0.975
	svm	0.925	0.945	0.979	0.962	0.985	0.98
	xgb	0.81	0.811	0.937	0.935	0.982	0.955
	mr	0.809	0.811	0.934	0.936	0.982	0.955
down	rf	0.824	0.873	0.977	0.928	0.92	0.95
	svm	0.803	0.839	0.977	0.905	0.915	0.97
	xgb	0.694	0.758	0.928	0.921	0.839	0.942
	mr	0.841	0.873	0.979	0.934	0.928	0.967
up	rf	0.928	0.958	0.982	0.957	0.983	0.976
	svm	0.916	0.944	0.979	0.955	0.978	0.977
	xgb	0.923	0.953	0.981	0.958	0.982	0.972
	mr	0.886	0.924	0.977	0.94	0.967	0.963
smote	rf	0.928	0.955	0.983	0.959	0.982	0.976
	svm	0.916	0.947	0.973	0.953	0.982	0.976
	xgb	0.927	0.957	0.98	0.959	0.985	0.972
	mr	0.901	0.935	0.982	0.949	0.969	0.967
hybrid	rf	0.917	0.95	0.976	0.953	0.981	0.975
	svm	0.916	0.943	0.979	0.953	0.979	0.977
	xgb	0.925	0.954	0.982	0.959	0.983	0.972
	mr	0.893	0.927	0.979	0.947	0.964	0.968

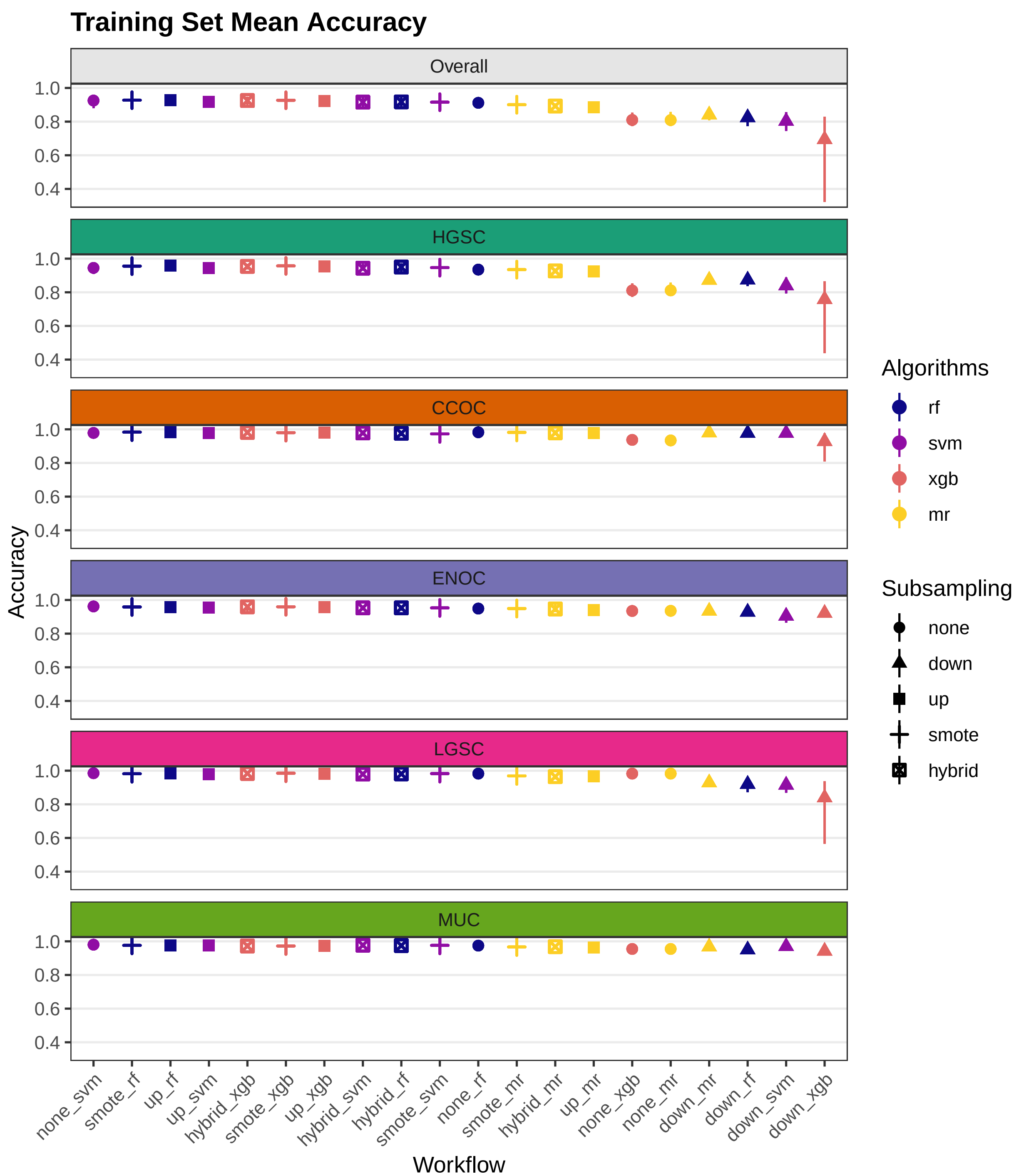

Figure 4.1: Training Set Mean Accuracy

4.1.2 Sensitivity

Table 4.2: Training Set Mean Sensitivity

			Histotypes				
Subsampling	Algorithms	Overall	HGSC	CCOC	ENOC	LGSC	MUC
none	rf	0.579	0.994	0.79	0.393	0	0.718
	svm	0.674	0.989	0.724	0.642	0.302	0.714
	xgb	0.208	1	0.04	0	0	0
	mr	0.207	1	0.013	0.022	0	0
down	rf	0.742	0.854	0.886	0.441	0.783	0.743
	svm	0.81	0.808	0.822	0.681	0.95	0.786
	xgb	0.693	0.701	0.873	0.4	0.855	0.636
	mr	0.815	0.851	0.861	0.689	0.855	0.822
up	rf	0.687	0.987	0.785	0.648	0.262	0.753
	svm	0.751	0.962	0.786	0.69	0.548	0.77
	xgb	0.761	0.967	0.819	0.633	0.548	0.839
	mr	0.766	0.922	0.81	0.671	0.648	0.776
smote	rf	0.712	0.979	0.833	0.646	0.312	0.788
	svm	0.744	0.967	0.74	0.646	0.598	0.77
	xgb	0.79	0.965	0.846	0.63	0.655	0.856
	mr	0.776	0.935	0.833	0.691	0.626	0.794
hybrid	rf	0.737	0.964	0.808	0.648	0.462	0.803
	svm	0.751	0.963	0.74	0.699	0.598	0.754
	xgb	0.79	0.964	0.846	0.646	0.655	0.839
	mr	0.796	0.924	0.833	0.657	0.755	0.81

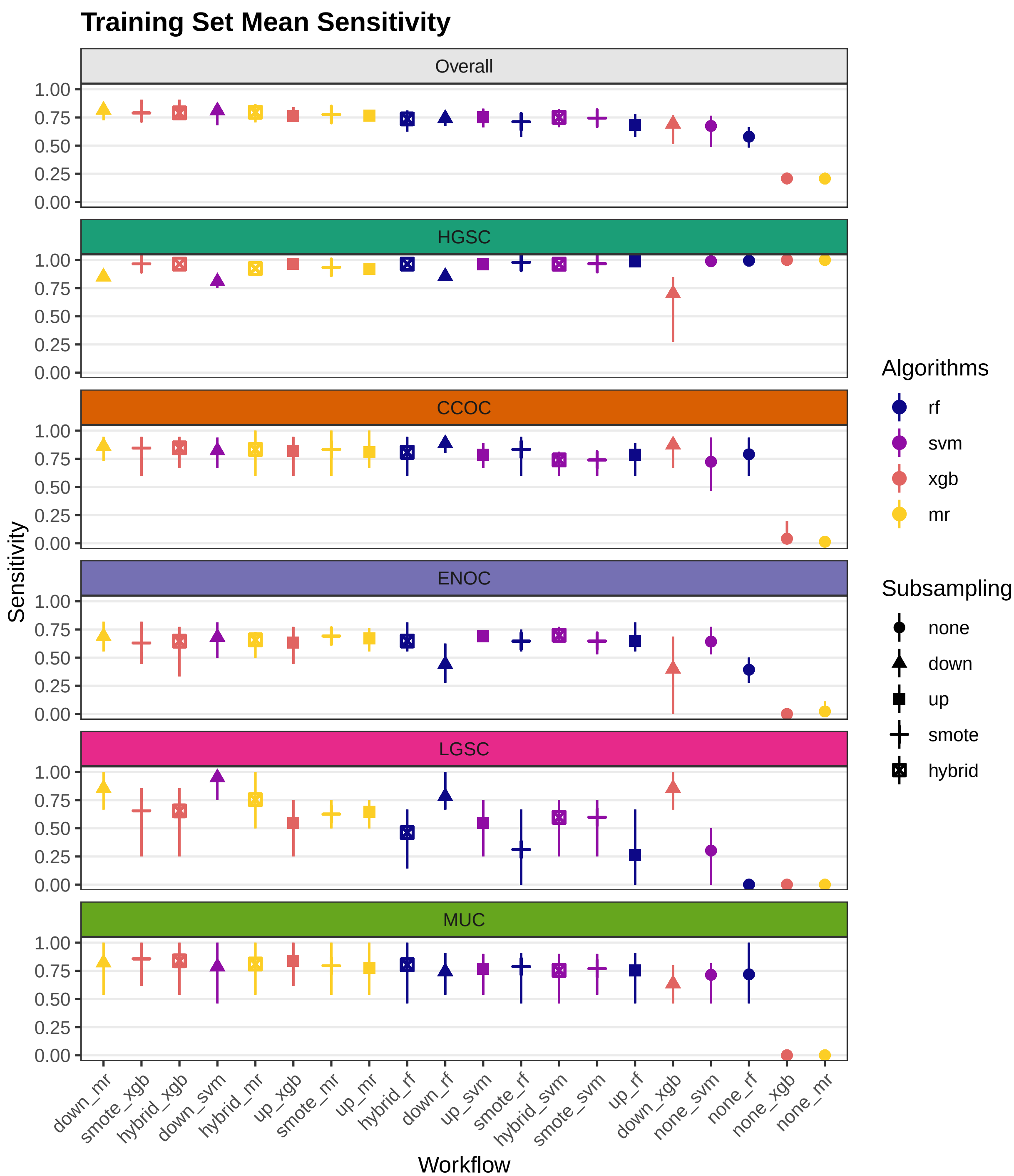

Figure 4.2: Training Set Mean Sensitivity

4.1.3 Specificity

Table 4.3: Training Set Mean Specificity

Subsampling	Algorithms	Overall	Histotypes HGSC	CCOC	ENOC	LGSC	MUC
none	rf	0.933	0.694	0.996	0.987	1	0.988
	svm	0.947	0.765	0.996	0.984	0.997	0.993
	xgb	0.803	0.016	0.999	1	1	1
	mr	0.804	0.021	0.997	1	1	1
down	rf	0.956	0.954	0.983	0.962	0.922	0.96
	svm	0.954	0.971	0.987	0.919	0.914	0.979
	xgb	0.932	0.974	0.932	0.96	0.838	0.957
	mr	0.961	0.962	0.987	0.95	0.93	0.975
up	rf	0.96	0.84	0.996	0.978	0.997	0.988
	svm	0.962	0.874	0.991	0.974	0.985	0.988
	xgb	0.967	0.897	0.991	0.98	0.99	0.979
	mr	0.966	0.935	0.988	0.959	0.973	0.973
smote	rf	0.963	0.861	0.993	0.98	0.994	0.986
	svm	0.96	0.863	0.989	0.974	0.99	0.987
	xgb	0.972	0.921	0.989	0.981	0.99	0.978
	mr	0.968	0.933	0.991	0.967	0.975	0.976
hybrid	rf	0.966	0.893	0.987	0.974	0.991	0.983
	svm	0.96	0.862	0.995	0.97	0.986	0.988
	xgb	0.97	0.91	0.991	0.981	0.989	0.979
	mr	0.967	0.938	0.989	0.967	0.968	0.977

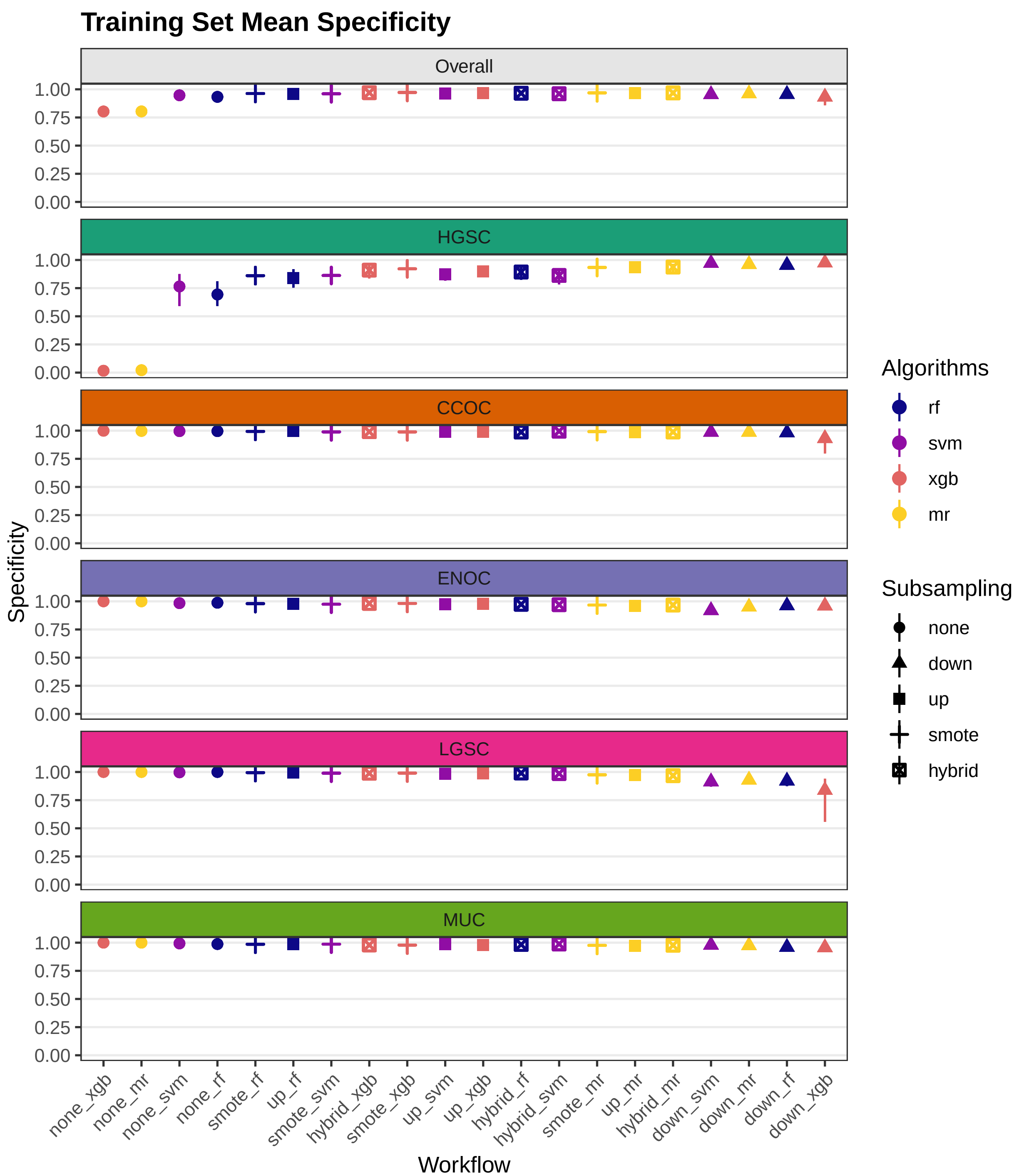

Figure 4.3: Training Set Mean Specificity

4.1.4 F1-Score

Table 4.4: Training Set Mean F1-Score

			Histotypes				
Subsampling	Algorithms	Overall	HGSC	CCOC	ENOC	LGSC	MUC
none	rf	0.752	0.961	0.848	0.487	NaN	0.713
	svm	0.723	0.967	0.8	0.673	0.413	0.762
	xgb	0.749	0.895	0.167	NaN	NaN	NaN
	mr	0.569	0.895	0.042	0.2	NaN	NaN
down	rf	0.605	0.916	0.832	0.433	0.27	0.574
	svm	0.635	0.89	0.82	0.478	0.292	0.698
	xgb	0.511	0.798	0.661	0.425	0.197	0.497
	mr	0.661	0.915	0.844	0.563	0.293	0.692
up	rf	0.736	0.974	0.846	0.652	0.392	0.734
	svm	0.729	0.965	0.822	0.661	0.448	0.751
	xgb	0.736	0.971	0.84	0.648	0.489	0.73
	mr	0.683	0.952	0.815	0.59	0.403	0.657
smote	rf	0.747	0.972	0.858	0.663	0.421	0.742
	svm	0.73	0.967	0.779	0.637	0.521	0.745
	xgb	0.755	0.973	0.84	0.654	0.576	0.733
	mr	0.708	0.959	0.848	0.633	0.417	0.682
hybrid	rf	0.718	0.968	0.809	0.632	0.449	0.732
	svm	0.731	0.965	0.813	0.65	0.482	0.746
	xgb	0.753	0.971	0.852	0.659	0.55	0.729
	mr	0.703	0.953	0.832	0.615	0.422	0.695

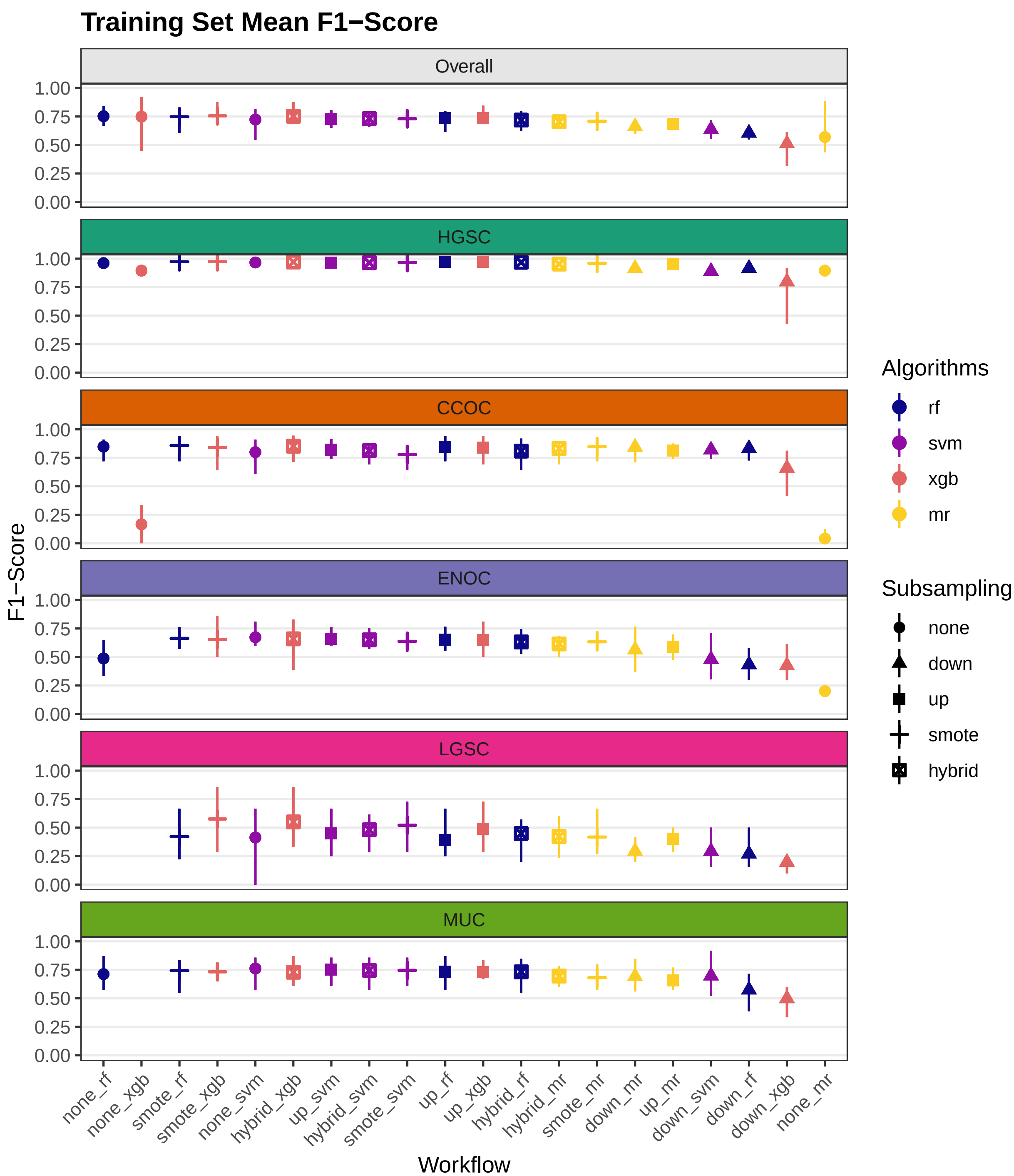

Figure 4.4: Training Set Mean F1-Score

4.1.5 Balanced Accuracy

Table 4.5: Training Set Mean Balanced Accuracy

Subsampling	Algorithms	Overall	Histotypes HGSC	CCOC	ENOC	LGSC	MUC
none	rf	0.756	0.844	0.893	0.69	0.5	0.853
	svm	0.811	0.877	0.86	0.813	0.65	0.854
	xgb	0.506	0.508	0.52	0.5	0.5	0.5
	mr	0.505	0.511	0.505	0.511	0.5	0.5
down	rf	0.849	0.904	0.934	0.702	0.852	0.852
	svm	0.882	0.89	0.905	0.8	0.932	0.883
	xgb	0.813	0.838	0.902	0.68	0.846	0.796
	mr	0.888	0.906	0.924	0.819	0.892	0.898
up	rf	0.823	0.913	0.891	0.813	0.629	0.87
	svm	0.857	0.918	0.889	0.832	0.767	0.879
	xgb	0.864	0.932	0.905	0.806	0.769	0.909
	mr	0.866	0.928	0.899	0.815	0.81	0.875
smote	rf	0.837	0.92	0.913	0.813	0.653	0.887
	svm	0.852	0.915	0.864	0.81	0.794	0.878
	xgb	0.881	0.943	0.917	0.806	0.823	0.917
	mr	0.872	0.934	0.912	0.829	0.801	0.885
hybrid	rf	0.851	0.929	0.897	0.811	0.726	0.893
	svm	0.856	0.913	0.867	0.835	0.792	0.871
	xgb	0.88	0.937	0.919	0.814	0.822	0.909
	mr	0.882	0.931	0.911	0.812	0.861	0.893

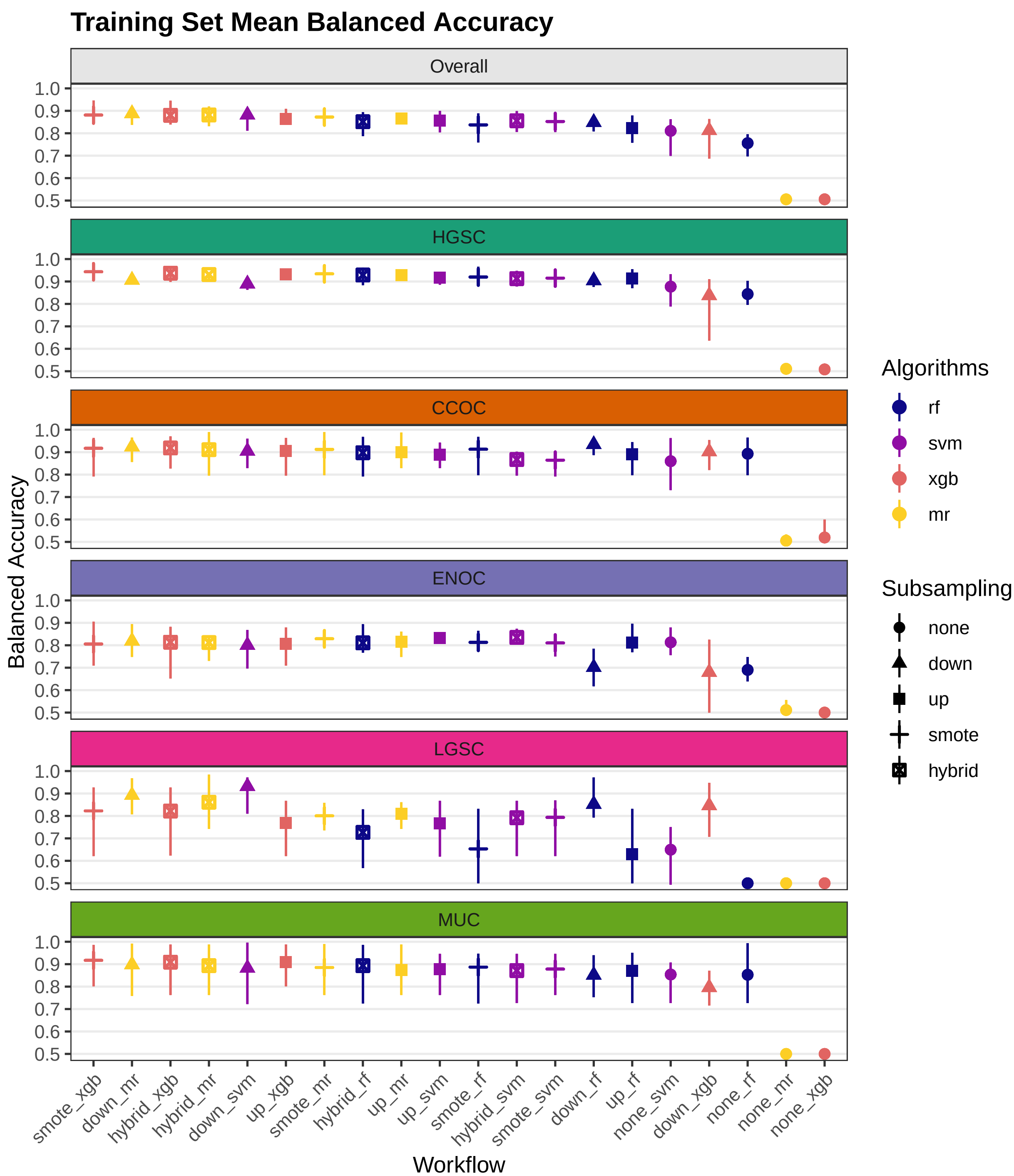

Figure 4.5: Training Set Mean Balanced Accuracy

4.1.6 Kappa

Table 4.6: Training Set Mean Kappa

			Histotypes				
Subsampling	Algorithms	Overall	HGSC	CCOC	ENOC	LGSC	MUC
none	rf	0.7	0.768	0.839	0.463	0	0.7
	svm	0.754	0.808	0.789	0.653	0.407	0.752
	xgb	0.023	0.026	0.062	0	0	0
	mr	0.025	0.034	0.019	0.039	0	0
down	rf	0.582	0.663	0.82	0.395	0.249	0.55
	svm	0.565	0.602	0.807	0.432	0.271	0.682
	xgb	0.447	0.501	0.628	0.308	0.171	0.469
	mr	0.623	0.663	0.833	0.529	0.273	0.675
up	rf	0.778	0.861	0.837	0.629	0.308	0.722
	svm	0.754	0.822	0.81	0.637	0.437	0.739
	xgb	0.773	0.85	0.83	0.625	0.481	0.716
	mr	0.695	0.777	0.802	0.558	0.389	0.638
smote	rf	0.78	0.856	0.849	0.641	0.33	0.73
	svm	0.749	0.829	0.764	0.612	0.512	0.733
	xgb	0.788	0.866	0.83	0.632	0.569	0.719
	mr	0.727	0.803	0.838	0.606	0.404	0.665
hybrid	rf	0.759	0.844	0.797	0.607	0.44	0.719
	svm	0.751	0.82	0.802	0.625	0.472	0.734
	xgb	0.782	0.854	0.842	0.638	0.543	0.715
	mr	0.711	0.784	0.821	0.586	0.408	0.679

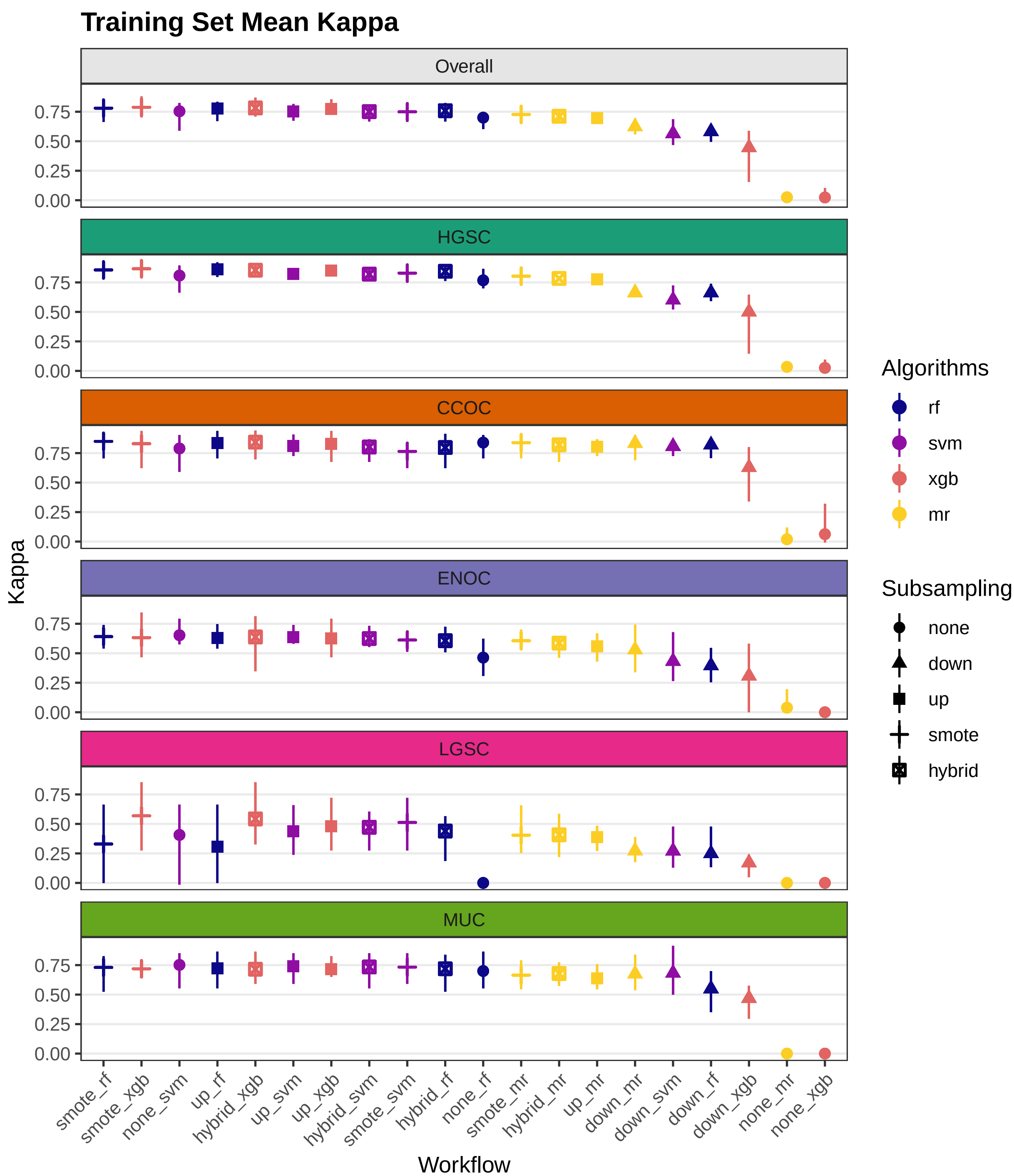

Figure 4.6: Training Set Mean Kappa

4.2 Rank Aggregation

Multi-step methods:

- **sequential**: sequential algorithm sequence of subsampling methods and algorithms used are:

- HGSC vs. non-HGSC using upsubsampling and random forest
- CCOC vs. non-CCOC using SMOTE subsampling and XGBoost
- ENOC vs. non-ENOC using hybrid subsampling and support vector machine
- LGSC vs. MUC using hybrid subsampling and random forest

- **two_step**: two-step algorithm sequence of subsampling methods and algorithms used are:
  - HGSC vs. non-HGSC using SMOTE subsampling and random forest
  - CCOC vs. ENOC vs. MUC vs. LGSC using hybrid subsampling and support vector machine

We conduct rank aggregation using a two-stage nested appraoch:

1. First we rank aggregate the per-class metrics for F1-score, balanced accuracy and kappa.
2. Then we take the aggregated lists from the three metrics and perform a final rank aggregation.
3. The top workflows from the final rank aggregation are used for gene optimization in the confirmation set

4.2.1 Across Classes

4.2.1.1 F1-Score

Table 4.7: F1-Score Rank Aggregation Summary

Workflow	Rank	HGSC	CCOC	ENOC	LGSC	MUC
All	All	All	All	All	All	All
sequential	1	0.97	0.891	0.852	0.92	0.963
two_step	2	0.969	0.865	0.738	0.782	0.864
smote_rf	3	0.972	0.858	0.663	0.421	0.742
hybrid_xgb	4	0.971	0.852	0.659	0.55	0.729
smote_xgb	5	0.973	0.84	0.654	0.576	0.733
up_rf	6	0.974	0.846	0.652	0.392	0.734
hybrid_svm	7	0.965	0.813	0.65	0.482	0.746
up_svm	8	0.965	0.822	0.661	0.448	0.751
smote_svm	9	0.967	0.779	0.637	0.521	0.745
up_xgb	10	0.971	0.84	0.648	0.489	0.73
hybrid_rf	11	0.968	0.809	0.632	0.449	0.732
none_svm	12	0.967	0.8	0.673	0.413	0.762
smote_mr	13	0.959	0.848	0.633	0.417	0.682
hybrid_mr	14	0.953	0.832	0.615	0.422	0.695
up_mr	15	0.952	0.815	0.59	0.403	0.657
down_mr	16	0.915	0.844	0.563	0.293	0.692
down_svm	17	0.89	0.82	0.478	0.292	0.698
down_rf	18	0.916	0.832	0.433	0.27	0.574
down_xgb	19	0.798	0.661	0.425	0.197	0.497

4.2.1.2 Balanced Accuracy

Table 4.8: Balanced Accuracy Rank Aggregation Summary

Workflow	Rank	HGSC	CCOC	ENOC	LGSC	MUC
All	All	All	All	All	All	All
sequential	1	0.913	0.913	0.858	0.953	0.953
smote_xgb	2	0.943	0.917	0.806	0.823	0.917
hybrid_xgb	3	0.937	0.919	0.814	0.822	0.909
smote_mr	4	0.934	0.912	0.829	0.801	0.885
down_mr	5	0.906	0.924	0.819	0.892	0.898
two_step	6	0.919	0.893	0.819	0.924	0.908
up_xgb	7	0.932	0.905	0.806	0.769	0.909
hybrid_mr	8	0.931	0.911	0.812	0.861	0.893
smote_rf	9	0.92	0.913	0.813	0.653	0.887
up_mr	10	0.928	0.899	0.815	0.81	0.875
down_svm	11	0.89	0.905	0.8	0.932	0.883
up_svm	12	0.918	0.889	0.832	0.767	0.879
hybrid_rf	13	0.929	0.897	0.811	0.726	0.893
smote_svm	14	0.915	0.864	0.81	0.794	0.878
hybrid_svm	15	0.913	0.867	0.835	0.792	0.871
up_rf	16	0.913	0.891	0.813	0.629	0.87
down_rf	17	0.904	0.934	0.702	0.852	0.852
none_svm	18	0.877	0.86	0.813	0.65	0.854
none_rf	19	0.844	0.893	0.69	0.5	0.853
down_xgb	20	0.838	0.902	0.68	0.846	0.796
none_mr	21	0.511	0.505	0.511	0.5	0.5
none_xgb	22	0.508	0.52	0.5	0.5	0.5

4.2.1.3 Kappa

Table 4.9: Kappa Rank Aggregation Summary

Workflow	Rank	HGSC	CCOC	ENOC	LGSC	MUC
All	All	All	All	All	All	All
sequential	1	0.842	0.839	0.715	0.884	0.884
smote_rf	2	0.856	0.849	0.641	0.33	0.73
smote_xgb	3	0.866	0.83	0.632	0.569	0.719
hybrid_xgb	4	0.854	0.842	0.638	0.543	0.715
two_step	5	0.833	0.796	0.632	0.758	0.818
up_svm	6	0.822	0.81	0.637	0.437	0.739
up_xgb	7	0.85	0.83	0.625	0.481	0.716
up_rf	8	0.861	0.837	0.629	0.308	0.722
smote_svm	9	0.829	0.764	0.612	0.512	0.733
hybrid_svm	10	0.82	0.802	0.625	0.472	0.734
hybrid_rf	11	0.844	0.797	0.607	0.44	0.719
none_svm	12	0.808	0.789	0.653	0.407	0.752
smote_mr	13	0.803	0.838	0.606	0.404	0.665
hybrid_mr	14	0.784	0.821	0.586	0.408	0.679
up_mr	15	0.777	0.802	0.558	0.389	0.638
down_mr	16	0.663	0.833	0.529	0.273	0.675
none_rf	17	0.768	0.839	0.463	0	0.7
down_svm	18	0.602	0.807	0.432	0.271	0.682
down_rf	19	0.663	0.82	0.395	0.249	0.55
down_xgb	20	0.501	0.628	0.308	0.171	0.469
none_mr	21	0.034	0.019	0.039	0	0
none_xgb	22	0.026	0.062	0	0	0

4.2.2 Across Metrics

Table 4.10: Rank Aggregation Comparison of Metrics Used

Rank	F1	Balanced Accuracy	Kappa
1	sequential	sequential	sequential
2	two_step	smote_xgb	smote_rf
3	smote_rf	hybrid_xgb	smote_xgb
4	hybrid_xgb	smote_mr	hybrid_xgb
5	smote_xgb	down_mr	two_step
6	up_rf	two_step	up_svm
7	hybrid_svm	up_xgb	up_xgb
8	up_svm	hybrid_mr	up_rf
9	smote_svm	smote_rf	smote_svm
10	up_xgb	up_mr	hybrid_svm
11	hybrid_rf	down_svm	hybrid_rf
12	none_svm	up_svm	none_svm
13	smote_mr	hybrid_rf	smote_mr
14	hybrid_mr	smote_svm	hybrid_mr
15	up_mr	hybrid_svm	up_mr
16	down_mr	up_rf	down_mr
17	down_svm	down_rf	none_rf
18	down_rf	none_svm	down_svm
19	down_xgb	none_rf	down_rf
20	NA	down_xgb	down_xgb
21	NA	none_mr	none_mr
22	NA	none_xgb	none_xgb

Table 4.11: Top 5 Workflows from Final Rank Aggregation

Rank	Workflow
1	sequential
2	smote_rf
3	smote_xgb
4	hybrid_xgb
5	two_step

4.2.3 Top Workflows

We look at the per-class evaluation metrics of the top 5 workflows.

Table 4.12: Top Workflow Per-Class Evaluation Metrics

		Histotypes				
Metric	Workflow	HGSC	CCOC	ENOC	LGSC	MUC
Accuracy	sequential	0.951 (0.94, 0.964)	0.929 (0.875, 0.96)	0.857 (0.781, 0.935)	0.95 (0.867, 1)	0.95 (0.867, 1)
	smote_rf	0.955 (0.936, 0.98)	0.983 (0.972, 0.988)	0.959 (0.94, 0.972)	0.982 (0.972, 0.992)	0.976 (0.96, 0.984)
	smote_xgb	0.957 (0.936, 0.98)	0.98 (0.96, 0.992)	0.959 (0.937, 0.976)	0.985 (0.976, 0.992)	0.972 (0.968, 0.98)
	hybrid_xgb	0.954 (0.936, 0.968)	0.982 (0.968, 0.992)	0.959 (0.925, 0.972)	0.983 (0.972, 0.992)	0.972 (0.964, 0.988)
	two_step	0.949 (0.924, 0.964)	0.909 (0.826, 0.957)	0.848 (0.783, 0.936)	0.957 (0.935, 0.978)	0.931 (0.891, 0.957)
Sensitivity	sequential	0.975 (0.961, 0.99)	0.863 (0.75, 0.941)	0.817 (0.75, 0.938)	0.96 (0.8, 1)	0.947 (0.818, 1)
	smote_rf	0.979 (0.969, 0.986)	0.833 (0.6, 0.944)	0.646 (0.556, 0.75)	0.312 (0, 0.667)	0.788 (0.462, 0.909)
	smote_xgb	0.965 (0.951, 0.986)	0.846 (0.6, 0.944)	0.63 (0.444, 0.818)	0.655 (0.25, 0.857)	0.856 (0.615, 1)
	hybrid_xgb	0.964 (0.956, 0.972)	0.846 (0.667, 0.944)	0.646 (0.333, 0.773)	0.655 (0.25, 0.857)	0.839 (0.538, 1)
	two_step	0.967 (0.95, 0.98)	0.839 (0.688, 0.933)	0.754 (0.583, 1)	0.883 (0.667, 1)	0.856 (0.786, 1)
Specificity	sequential	0.851 (0.833, 0.875)	0.963 (0.938, 1)	0.899 (0.812, 0.938)	0.947 (0.818, 1)	0.96 (0.8, 1)
	smote_rf	0.861 (0.776, 0.946)	0.993 (0.987, 0.996)	0.98 (0.966, 0.988)	0.994 (0.984, 1)	0.986 (0.979, 0.996)
	smote_xgb	0.921 (0.857, 0.965)	0.989 (0.983, 1)	0.981 (0.97, 0.991)	0.99 (0.98, 0.996)	0.978 (0.971, 0.988)
	hybrid_xgb	0.91 (0.837, 0.947)	0.991 (0.987, 0.996)	0.981 (0.97, 0.991)	0.989 (0.976, 0.996)	0.979 (0.967, 0.992)
	two_step	0.871 (0.766, 0.957)	0.947 (0.9, 1)	0.884 (0.812, 1)	0.966 (0.921, 1)	0.96 (0.917, 1)
F1-Score	sequential	0.97 (0.963, 0.978)	0.891 (0.8, 0.941)	0.852 (0.774, 0.938)	0.92 (0.8, 1)	0.963 (0.9, 1)
	smote_rf	0.972 (0.959, 0.988)	0.858 (0.72, 0.919)	0.663 (0.571, 0.762)	0.421 (0.222, 0.667)	0.742 (0.545, 0.833)
	smote_xgb	0.973 (0.96, 0.988)	0.84 (0.643, 0.941)	0.654 (0.5, 0.857)	0.576 (0.286, 0.857)	0.733 (0.667, 0.783)
	hybrid_xgb	0.971 (0.96, 0.981)	0.852 (0.714, 0.944)	0.659 (0.387, 0.829)	0.55 (0.333, 0.857)	0.729 (0.609, 0.87)
	two_step	0.969 (0.954, 0.978)	0.865 (0.733, 0.941)	0.738 (0.615, 0.897)	0.782 (0.667, 0.842)	0.864 (0.762, 0.917)
Balanced Accuracy	sequential	0.913 (0.899, 0.93)	0.913 (0.844, 0.955)	0.858 (0.781, 0.935)	0.953 (0.9, 1)	0.953 (0.9, 1)
	smote_rf	0.92 (0.878, 0.966)	0.913 (0.798, 0.968)	0.813 (0.772, 0.864)	0.653 (0.5, 0.831)	0.887 (0.724, 0.946)
	smote_xgb	0.943 (0.906, 0.97)	0.917 (0.792, 0.964)	0.806 (0.709, 0.905)	0.823 (0.621, 0.927)	0.917 (0.801, 0.986)
	hybrid_xgb	0.937 (0.899, 0.959)	0.919 (0.827, 0.97)	0.814 (0.652, 0.882)	0.822 (0.623, 0.927)	0.909 (0.763, 0.988)
	two_step	0.919 (0.863, 0.954)	0.893 (0.794, 0.951)	0.819 (0.745, 0.956)	0.924 (0.833, 0.989)	0.908 (0.858, 0.971)
Kappa	sequential	0.842 (0.805, 0.881)	0.839 (0.71, 0.911)	0.715 (0.562, 0.871)	0.884 (0.706, 1)	0.884 (0.706, 1)
	smote_rf	0.856 (0.799, 0.922)	0.849 (0.706, 0.912)	0.641 (0.539, 0.74)	0.33 (0, 0.663)	0.73 (0.525, 0.825)
	smote_xgb	0.866 (0.8, 0.922)	0.83 (0.622, 0.937)	0.632 (0.467, 0.844)	0.569 (0.276, 0.853)	0.719 (0.65, 0.772)
	hybrid_xgb	0.854 (0.797, 0.9)	0.842 (0.697, 0.94)	0.638 (0.348, 0.814)	0.543 (0.326, 0.853)	0.715 (0.59, 0.863)
	two_step	0.833 (0.745, 0.883)	0.796 (0.605, 0.908)	0.632 (0.465, 0.851)	0.758 (0.647, 0.802)	0.818 (0.692, 0.888)

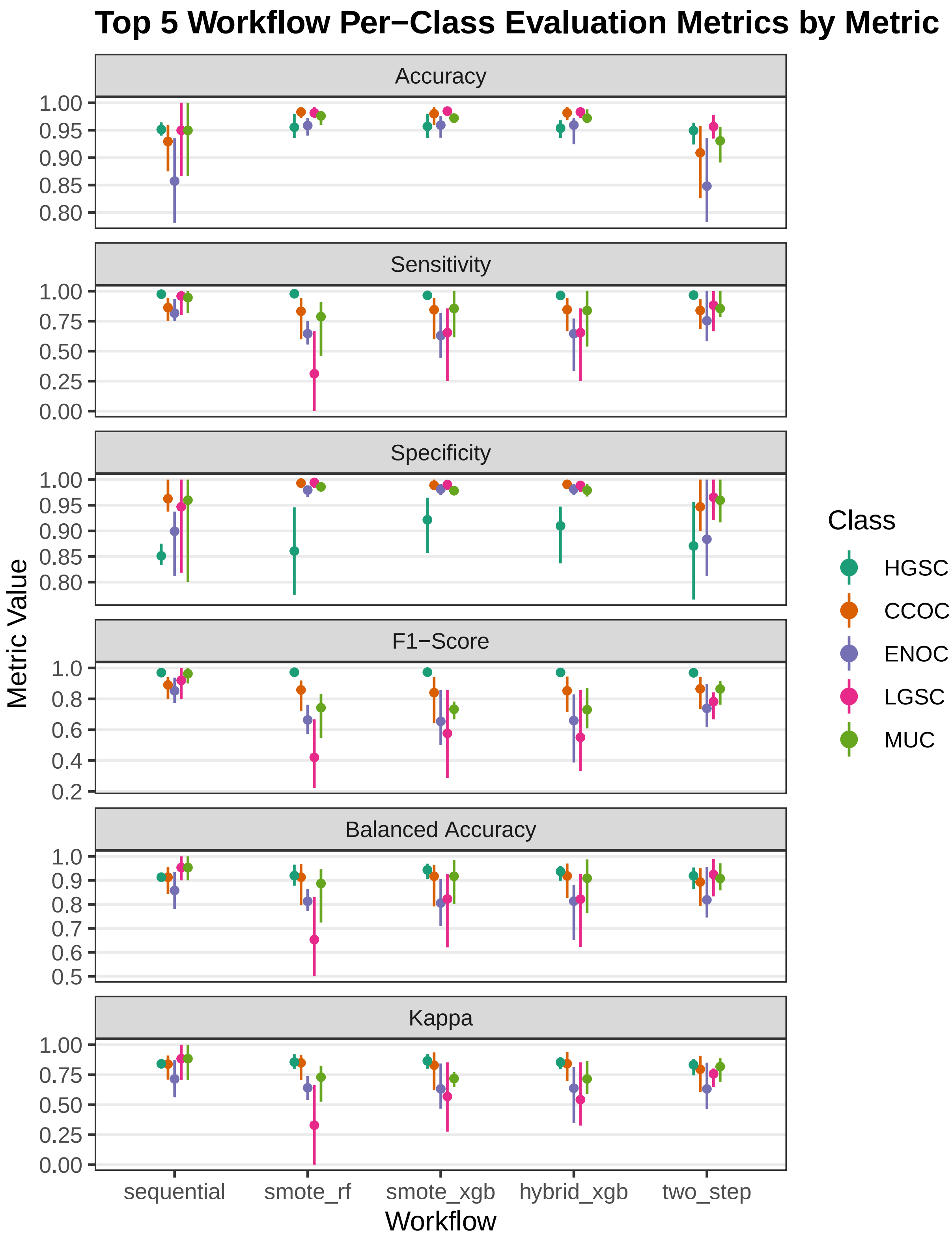

Figure 4.7: Top 5 Workflow Per-Class Evaluation Metrics by Metric

Table 4.13: Top Workflow Per-Class Evaluation Metrics and Ranks

Workflow	Rank	HGSC	CCOC	ENOC	LGSC	MUC
F1-Score						
sequential	1	0.970	0.891	0.852	0.920	0.963
two_step	2	0.969	0.865	0.738	0.782	0.864
smote_rf	3	0.972	0.858	0.663	0.421	0.742
hybrid_xgb	4	0.971	0.852	0.659	0.550	0.729
smote_xgb	5	0.973	0.840	0.654	0.576	0.733
Balanced Accuracy						
sequential	1	0.913	0.913	0.858	0.953	0.953
smote_xgb	2	0.943	0.917	0.806	0.823	0.917
hybrid_xgb	3	0.937	0.919	0.814	0.822	0.909
two_step	6	0.919	0.893	0.819	0.924	0.908
smote_rf	9	0.920	0.913	0.813	0.653	0.887
Kappa						
sequential	1	0.842	0.839	0.715	0.884	0.884
smote_rf	2	0.856	0.849	0.641	0.330	0.730
smote_xgb	3	0.866	0.830	0.632	0.569	0.719
hybrid_xgb	4	0.854	0.842	0.638	0.543	0.715
two_step	5	0.833	0.796	0.632	0.758	0.818

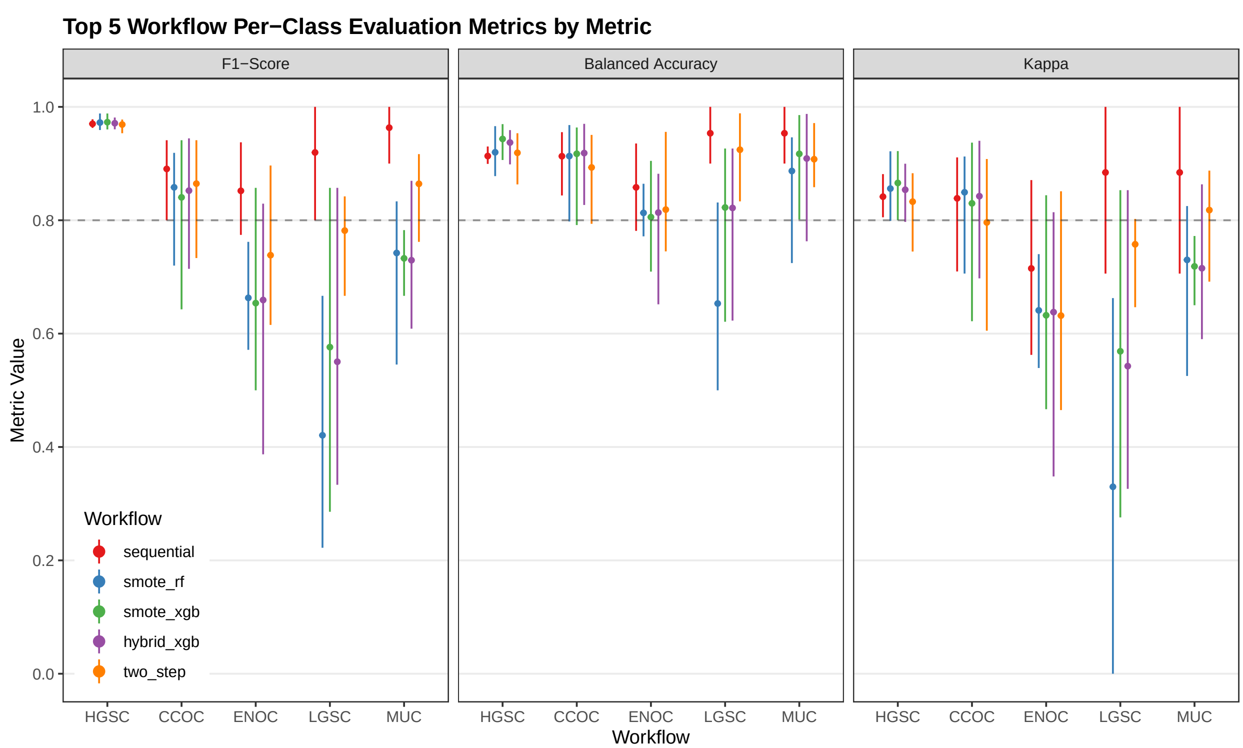

Figure 4.8: Top 5 Workflow Per-Class Evaluation Metrics by Metric

Misclassified cases from a previous step of the sequence of classifiers are not included in subsequent steps of the training set CV folds. Thus, we cannot piece together the test set predictions from the sequential and two-step algorithms to obtain overall metrics.

4.3 Optimal Gene Sets

4.3.1 Sequential Algorithm

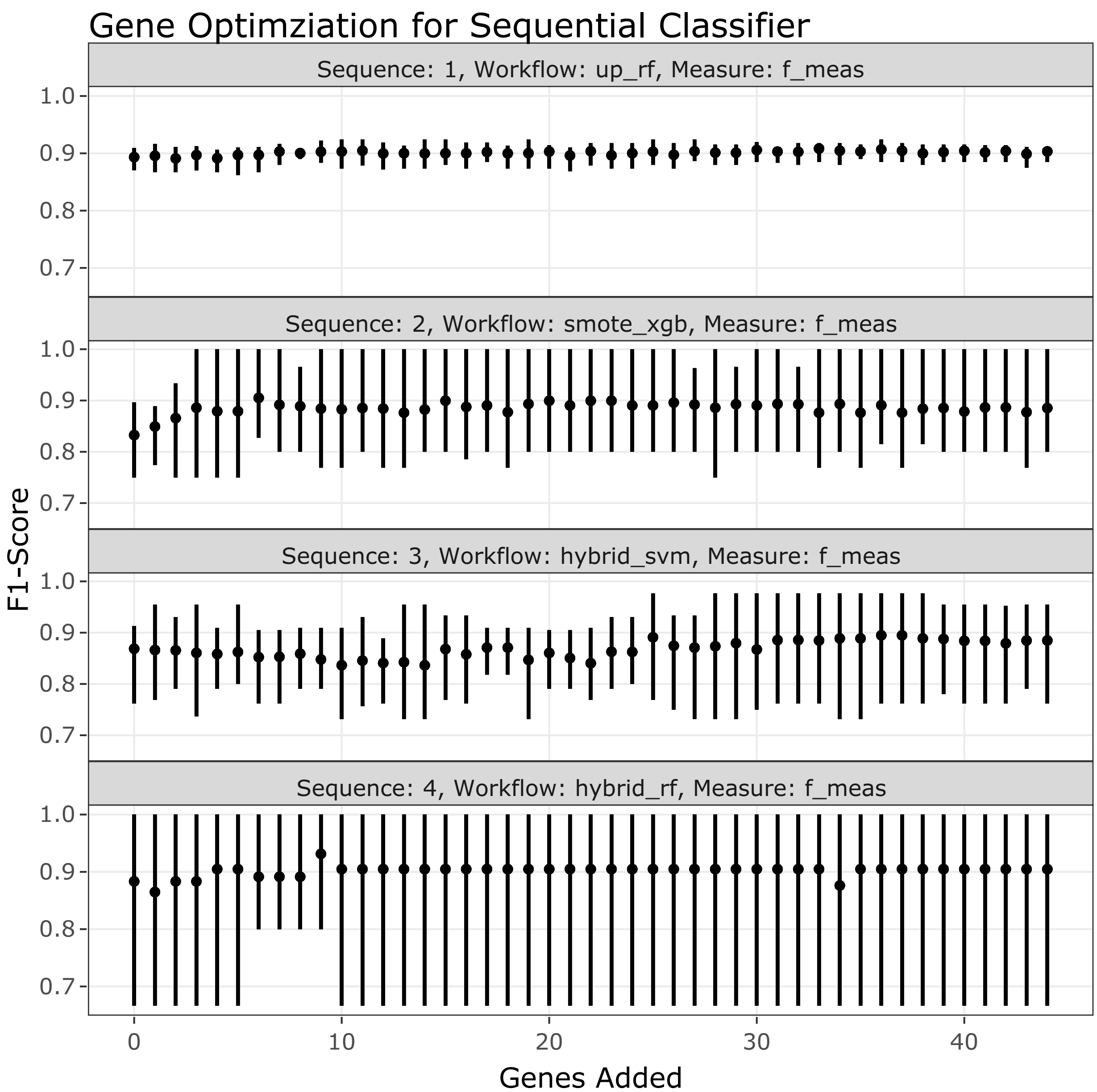

Figure 4.9: Gene Optimization for Sequential Classifier

In the sequential algorithm, all sequences have relatively flat average F1-scores across the number of genes added. However, we can observe in sequence 4, the F1-score is highest when we reach 9 genes added, hence the optimal number of genes used will be n=28+9=37 The added genes are: CYP2C18, HNF1B, EGFL6, TFF3, IL6, CYP4B1, LGALS4, SLC3A1 and IGFBP1.

Table 4.14: Gene Profile of Optimal Set in Sequential Algorithm

Set	Genes	PrOTYPE	SPOT	Optimal Set	Candidate Rank
	COL11A1	v		(*)	
	CD74	v		(*)	
	CD2	v		(*)	
	TIMP3	v		(*)	
	LUM	v		(*)	
	CYTIP	v		(*)	
	COL3A1	v		(*)	
	THBS2	v		(*)	
	TCF7L1	v	v	(*)	
	HMGA2	v		(*)	
	FN1	v		(*)	
	POSTN	v		(*)	
	COL1A2	v		(*)	
	COL5A2	v		(*)	
	PDZK1IP1	v		(*)	
	FBN1	v		(*)	
	HIF1A		v	(*)	
Base	CXCL10		v	(*)	
	DUSP4		v	(*)	
	SOX17		v	(*)	
	MITF		v	(*)	
	CDKN3		v	(*)	
	BRCA2		v	(*)	
	CEACAM5		v	(*)	
	ANXA4		v	(*)	
	SERPINE1		v	(*)	
	CRABP2		v	(*)	
	DNAJC9		v	(*)	

CYP2C18	(*)	1
HNF1B	(*)	2
EGFL6	(*)	3
TFF3	(*)	4
IL6	(*)	5
CYP4B1	(*)	6
LGALS4	(*)	7
SLC3A1	(*)	8
IGFBP1	(*)	9
WT1	(x)	10
MUC5B	(x)	11
TFF1	(x)	12
GPR64	(x)	13
TP53	(x)	14
BRCA1	(x)	15
MET	(x)	16
FUT3	(x)	17
CPNE8	(x)	18
TPX2	(x)	19
PBX1	(x)	20
EPAS1	(x)	21
SCGB1D2	(x)	22
KLK7	(x)	23
SEMA6A	(x)	24
DKK4	(x)	25
CAPN2	(x)	26
GAD1	(x)	27
STC1	(x)	28
IGJ	(x)	29
GCNT3	(x)	30
TSPAN8	(x)	31
SERPINA5	(x)	32
C1orf173	(x)	33
PAX8	(x)	34

LIN28B	(x)	35
ZBED1	(x)	36
ATP5G3	(x)	37
BCL2	(x)	38
KGFLP2	(x)	39
IGKC	(x)	40
SENP8	(x)	41
MAP1LC3A	(x)	42
C10orf116	(x)	43
ADCYAP1R1	(x)	44

4.3.2 SMOTE-Random Forest

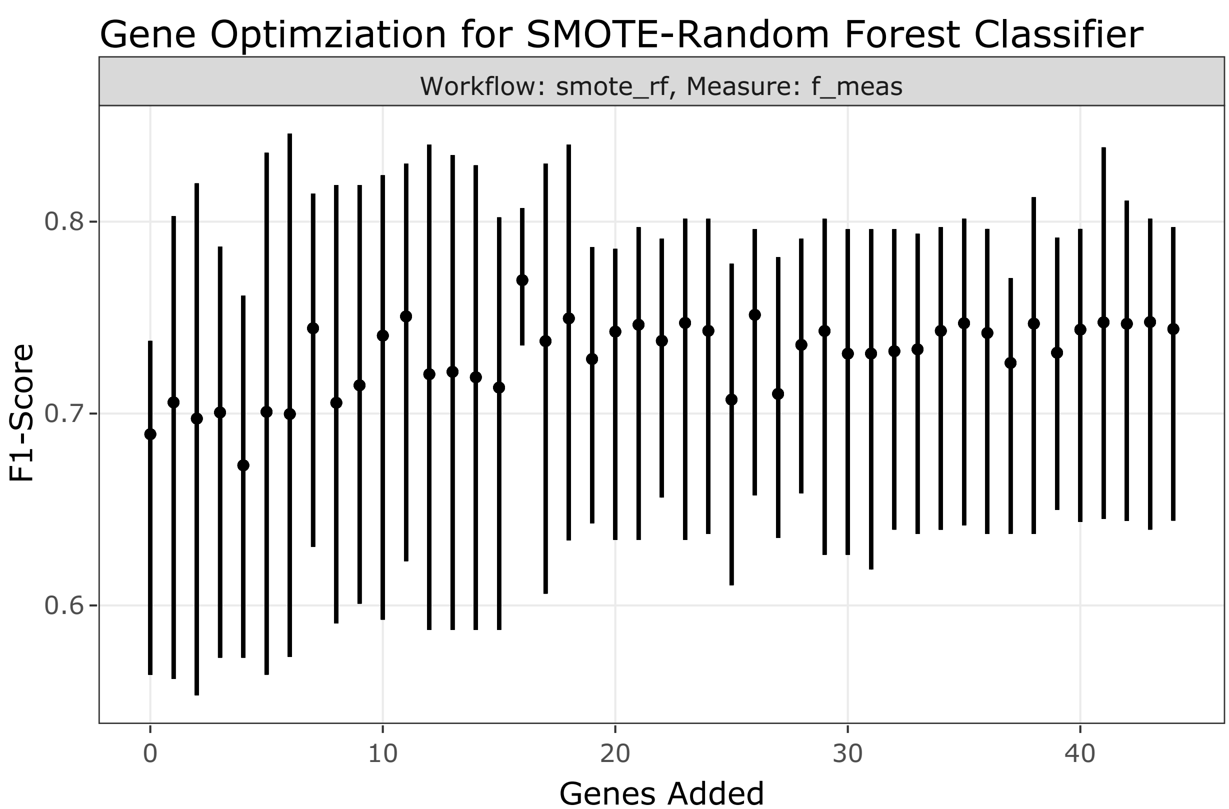

Figure 4.10: Gene Optimization for SMOTE-Random Forest Classifier

In the SMOTE-Random Forest classifier, the mean F1-score is highest when we reach 16 genes added, hence the optimal number of genes used will be n=28+16=44 The added genes are: HNF1B, TFF3, TPX2, SLC3A1, CYP2C18, TFF1, WT1, KLK7, IGFBP1, LGALS4, GAD1, GCNT3, C1orf173, CAPN2, FUT3 and DKK4.

Table 4.15: Gene Profile of Optimal Set in SMOTE-Random Forest Workflow

Set	Genes	PrOTYPE	SPOT	Optimal Set	Candidate Rank
Base	COL11A1	v		(*)	
	CD74	v		(*)	
	CD2	v		(*)	
	TIMP3	v		(*)	
	LUM	v		(*)	
	CYTIP	v		(*)	
	COL3A1	v		(*)	
	THBS2	v		(*)	
	TCF7L1	v	v	(*)	
	HMGA2	v		(*)	
	FN1	v		(*)	
	POSTN	v		(*)	
	COL1A2	v		(*)	
	COL5A2	v		(*)	
	PDZK1IP1	v		(*)	
	FBN1	v		(*)	
	HIF1A		v	(*)	
	CXCL10		v	(*)	
	DUSP4		v	(*)	
	SOX17		v	(*)	
	MITF		v	(*)	
	CDKN3		v	(*)	
	BRCA2		v	(*)	
	CEACAM5		v	(*)	
	ANXA4		v	(*)	
	SERPINE1		v	(*)	
	CRABP2		v	(*)	
	DNAJC9		v	(*)	
	HNF1B			(*)	1
	TFF3			(*)	2
	TPX2			(*)	3
	SLC3A1			(*)	4

CYP2C18	(*)	5
TFF1	(*)	6
WT1	(*)	7
KLK7	(*)	8
IGFBP1	(*)	9
LGALS4	(*)	10
GAD1	(*)	11
GCNT3	(*)	12
C1orf173	(*)	13
CAPN2	(*)	14
FUT3	(*)	15
DKK4	(*)	16
C10orf116	(x)	17
MUC5B	(x)	18
MET	(x)	19
GPR64	(x)	20
IGKC	(x)	21
PAX8	(x)	22
ATP5G3	(x)	23
CPNE8	(x)	24
PBX1	(x)	25
IL6	(x)	26
TP53	(x)	27
KGFLP2	(x)	28
EGFL6	(x)	29
SEMA6A	(x)	30
CYP4B1	(x)	31
STC1	(x)	32
EPAS1	(x)	33
BRCA1	(x)	34
LIN28B	(x)	35
TSPAN8	(x)	36
SERPINA5	(x)	37
SCGB1D2	(x)	38

BCL2	(x)	39
ZBED1	(x)	40
ADCYAP1R1	(x)	41
IGJ	(x)	42
SENP8	(x)	43
MAP1LC3A	(x)	44

4.3.3 Two-Step

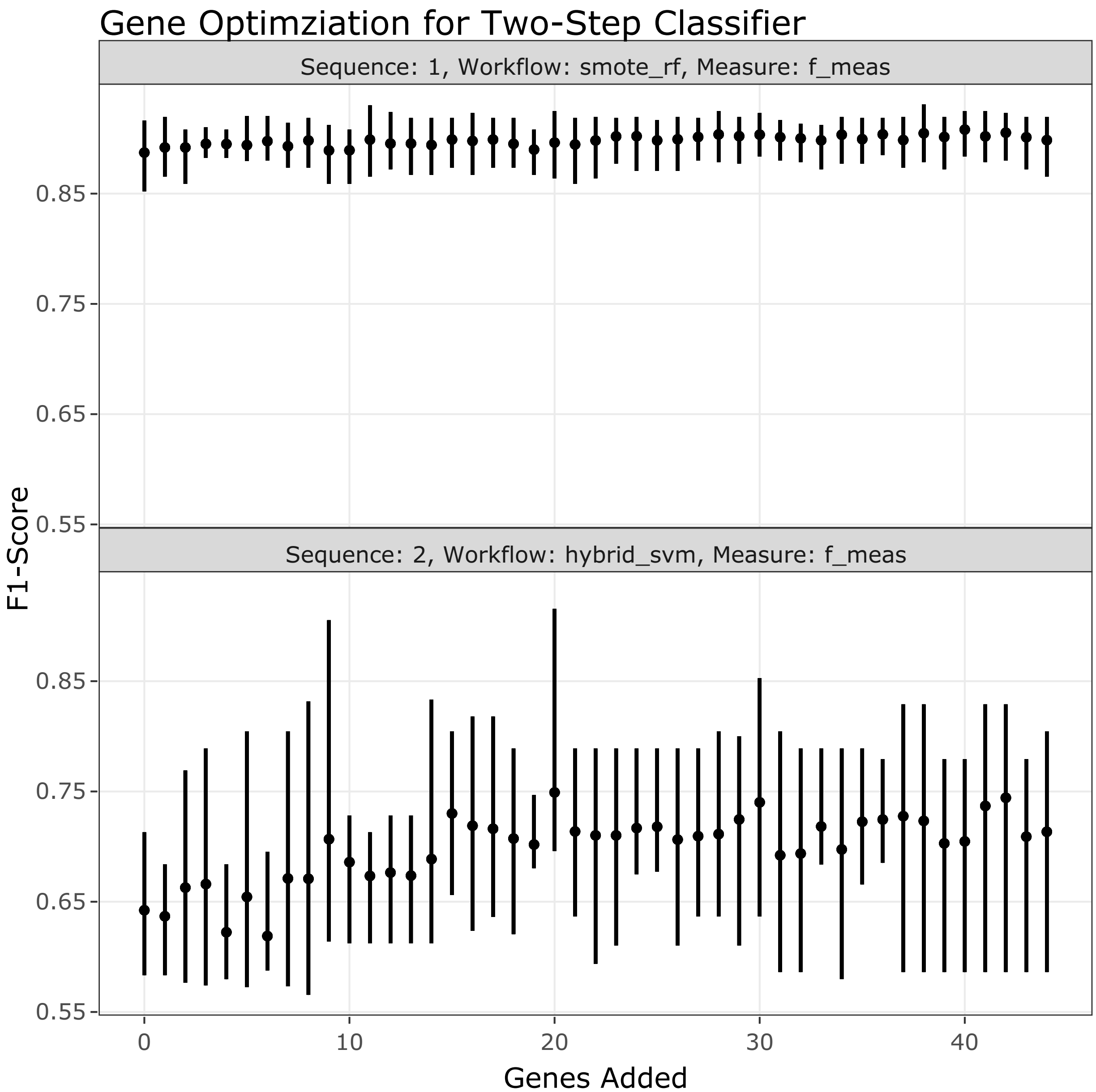

Figure 4.11: Gene Optimization for Two-Step Classifier

Table 4.16: Gene Profile of Optimal Set in Two-Step Workflow

Set	Genes	PrOTYPE	SPOT	Optimal Set	Candidate Rank
	COL11A1	v		(*)	
	CD74	v		(*)	
	CD2	v		(*)	

	TIMP3	v		(*)	
	LUM	v		(*)	
	CYTIP	v		(*)	
	COL3A1	v		(*)	
	THBS2	v		(*)	
	TCF7L1	v	v	(*)	
	HMGA2	v		(*)	
	FN1	v		(*)	
	POSTN	v		(*)	
	COL1A2	v		(*)	
	COL5A2	v		(*)	
	PDZK1IP1	v		(*)	
	FBN1	v		(*)	
	HIF1A		v	(*)	
Base	CXCL10		v	(*)	
	DUSP4		v	(*)	
	SOX17		v	(*)	
	MITF		v	(*)	
	CDKN3		v	(*)	
	BRCA2		v	(*)	
	CEACAM5		v	(*)	
	ANXA4		v	(*)	
	SERPINE1		v	(*)	
	CRABP2		v	(*)	
	DNAJC9		v	(*)	
	CYP2C18			(*)	1
	MUC5B			(*)	2
	HNF1B			(*)	3
	IL6			(*)	4
	SLC3A1			(*)	5
	EGFL6			(*)	6
	WT1			(*)	7
	ZBED1			(*)	8
	MET			(*)	9

SENP8	(*)	10
KLK7	(*)	11
TFF3	(*)	12
CPNE8	(*)	13
STC1	(*)	14
GAD1	(*)	15
LIN28B	(x)	16
IGJ	(x)	17
DKK4	(x)	18
EPAS1	(x)	19
GCNT3	(x)	20
SCGB1D2	(x)	21
CYP4B1	(x)	22
C1orf173	(x)	23
IGFBP1	(x)	24
TPX2	(x)	25
SEMA6A	(x)	26
ATP5G3	(x)	27
SERPINA5	(x)	28
FUT3	(x)	29
C10orf116	(x)	30
KGFLP2	(x)	31
ADCYAP1R1	(x)	32
TP53	(x)	33
PBX1	(x)	34
GPR64	(x)	35
LGALS4	(x)	36
CAPN2	(x)	37
BCL2	(x)	38
MAP1LC3A	(x)	39
TSPAN8	(x)	40
TFF1	(x)	41
PAX8	(x)	42
BRCA1	(x)	43

IGKC	(x)	44

4.4 Test Set Performance

Now we'd like to see how our best methods perform in the confirmation and validation sets. The class-specific F1-scores will be used.

The top 2 methods are the sequential and SMOTE-Random Forest classifiers. We can test 2 additional methods by using either the full set of genes or the optimal set of genes for both of these classifiers.

4.4.1 Confirmation Set

Table 4.17: Evaluation Metrics on Confirmation Set Models

			Histotypes				
Method	Metric	Overall	HGSC	CCOC	ENOC	LGSC	MUC
Sequential, Full Set	Accuracy	0.829	0.861	0.964	0.888	0.975	0.969
	Sensitivity	0.591	0.950	0.861	0.467	0.083	0.593
	Specificity	0.923	0.688	0.977	0.972	0.992	0.985
	F1-Score	0.610	0.901	0.844	0.581	0.111	0.615
	Balanced Accuracy	0.757	0.819	0.919	0.720	0.538	0.789
	Kappa	0.646	0.674	0.823	0.521	0.100	0.599
Sequential, Optimal Set	Accuracy	0.816	0.852	0.963	0.875	0.970	0.972
	Sensitivity	0.554	0.955	0.875	0.383	0.000	0.556
	Specificity	0.916	0.651	0.974	0.974	0.989	0.990
	F1-Score	0.573	0.895	0.840	0.506	0.000	0.625
	Balanced Accuracy	0.735	0.803	0.924	0.679	0.494	0.773
	Kappa	0.614	0.648	0.819	0.443	-0.014	0.611
SMOTE-Random Forest, Full Set	Accuracy	0.841	0.864	0.972	0.896	0.977	0.974
	Sensitivity	0.647	0.960	0.861	0.458	0.250	0.704
	Specificity	0.925	0.679	0.986	0.983	0.990	0.985
	F1-Score	0.669	0.903	0.873	0.594	0.286	0.691
	Balanced Accuracy	0.786	0.819	0.924	0.721	0.620	0.845
	Kappa	0.669	0.679	0.857	0.540	0.274	0.677
SMOTE-Random Forest, Optimal Set	Accuracy	0.838	0.868	0.967	0.896	0.980	0.966
	Sensitivity	0.659	0.948	0.861	0.486	0.333	0.667
	Specificity	0.928	0.711	0.981	0.978	0.992	0.979
	F1-Score	0.674	0.904	0.855	0.608	0.381	0.621
	Balanced Accuracy	0.794	0.830	0.921	0.732	0.663	0.823
	Kappa	0.669	0.691	0.837	0.552	0.371	0.603
Two-Step, Full Set	Accuracy	0.835	0.861	0.966	0.891	0.972	0.980
	Sensitivity	0.651	0.941	0.875	0.486	0.250	0.704
	Specificity	0.927	0.706	0.977	0.972	0.986	0.992
	F1-Score	0.669	0.900	0.851	0.598	0.250	0.745
	Balanced Accuracy	0.789	0.824	0.926	0.729	0.618	0.848
	Kappa	0.664	0.677	0.832	0.538	0.236	0.735
Two-Step, Optimal Set	Accuracy	0.843	0.866	0.967	0.900	0.975	0.977
	Sensitivity	0.639	0.953	0.875	0.495	0.167	0.704
	Specificity	0.927	0.697	0.979	0.981	0.990	0.989
	F1-Score	0.660	0.904	0.857	0.624	0.200	0.717
	Balanced Accuracy	0.783	0.825	0.927	0.738	0.579	0.846
	Kappa	0.676	0.685	0.839	0.570	0.188	0.705

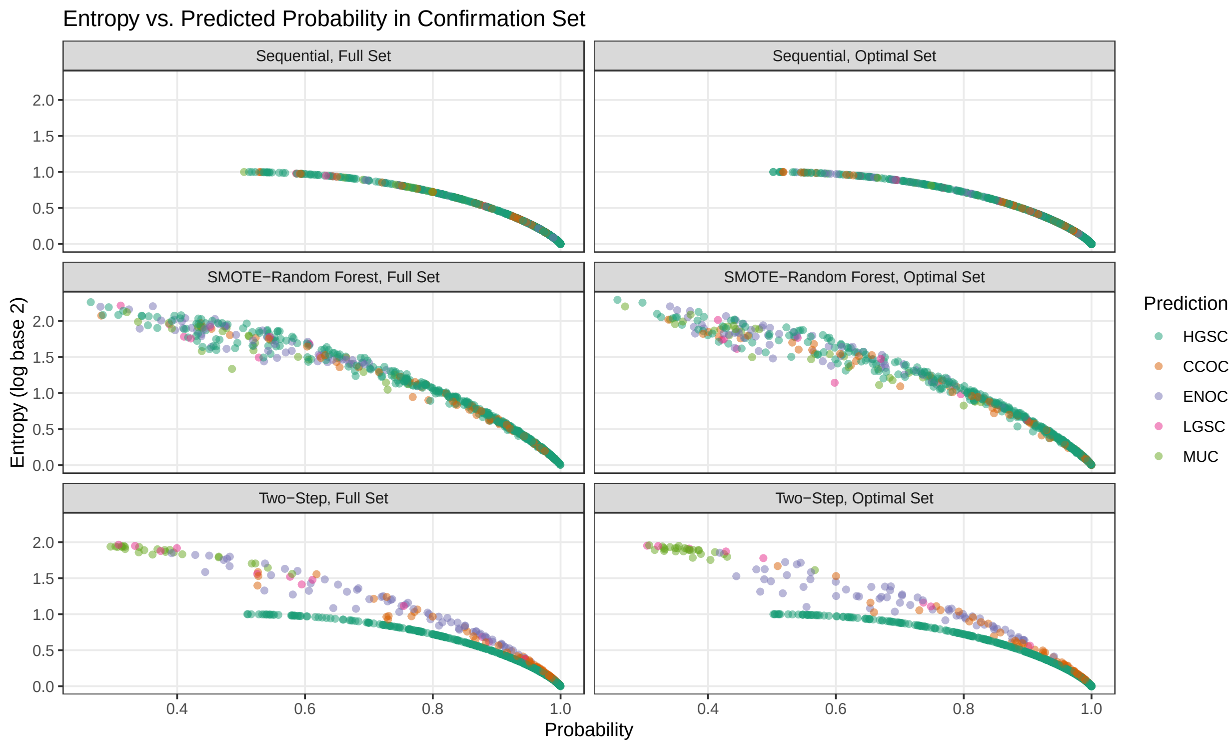

Figure 4.12: Entropy vs. Predicted Probability in Confirmation Set

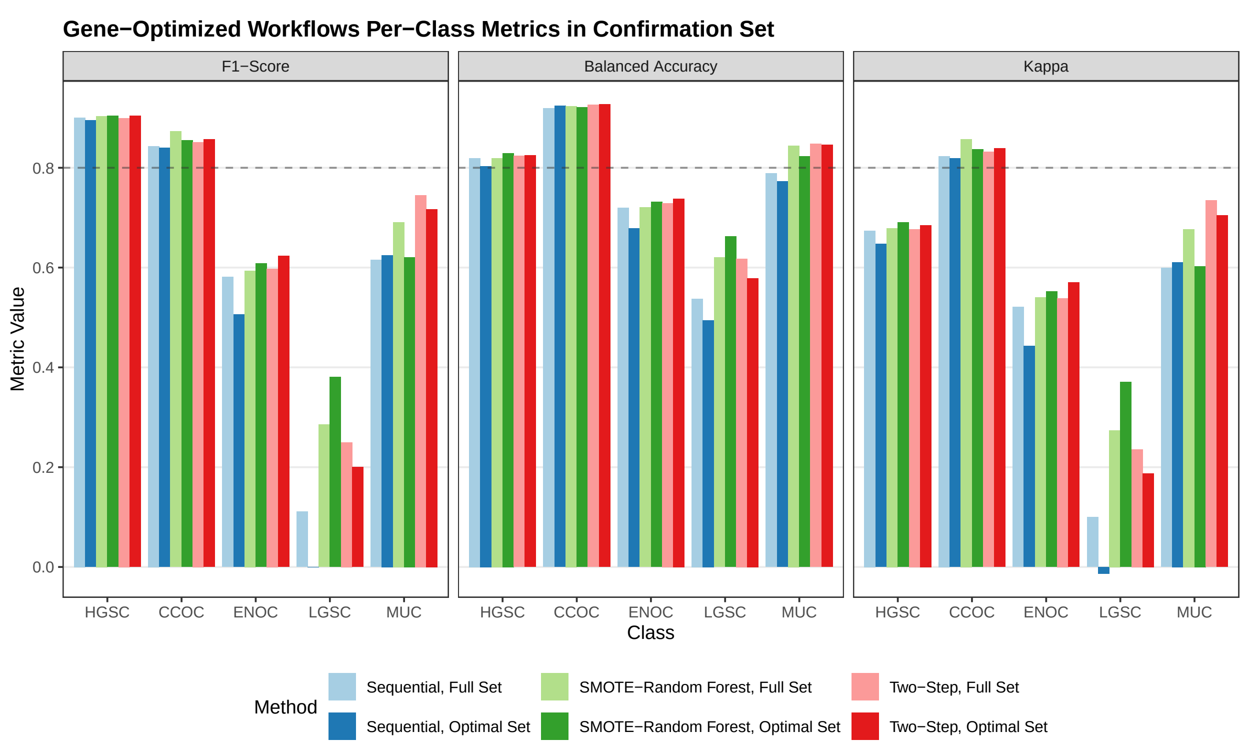

Figure 4.13: Gene Optimized Workflows Per-Class Metrics in Confirmation Set

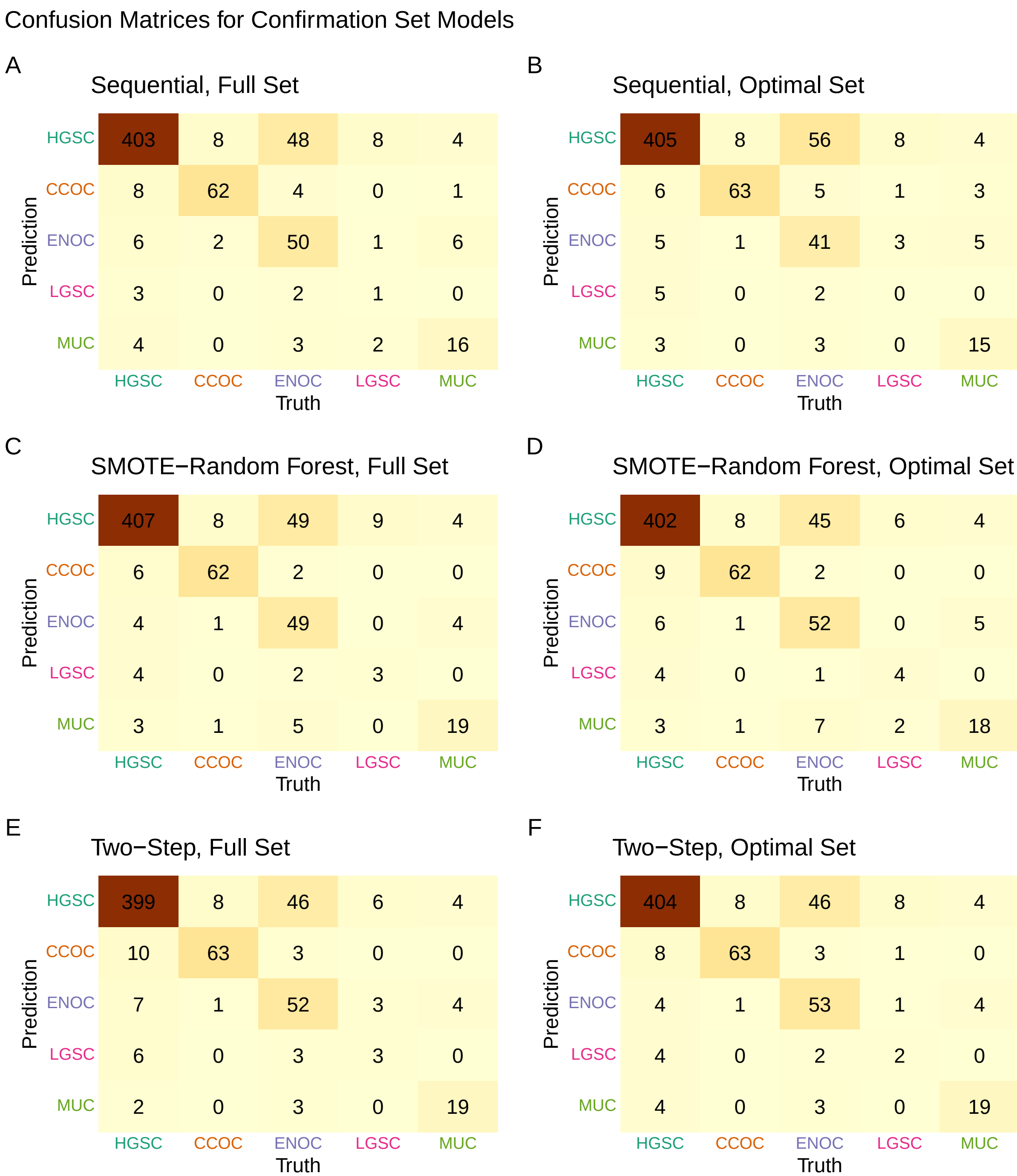

Figure 4.14: Confusion Matrices for Confirmation Set Models

4.4.1.1 Sequential, Full

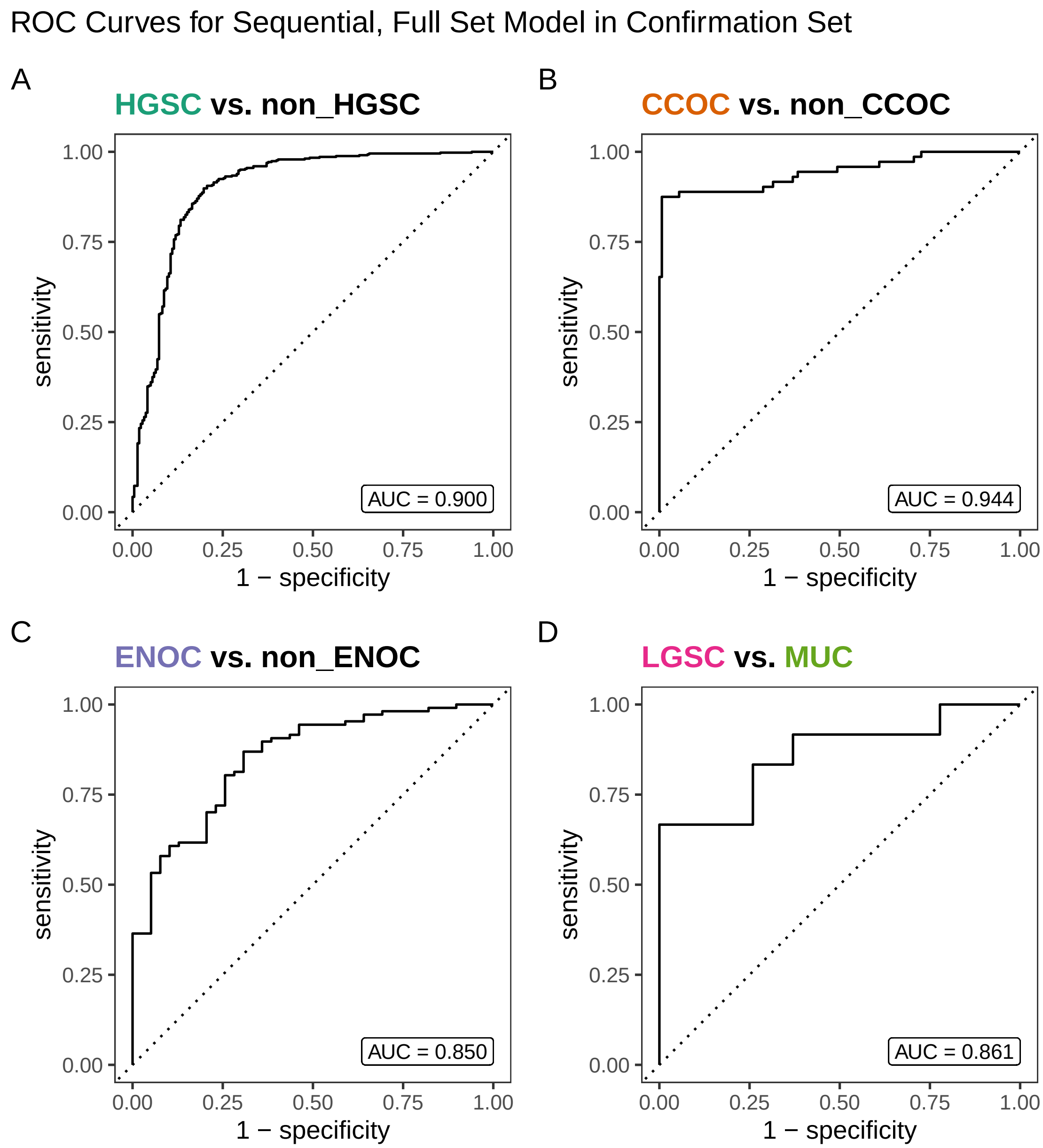

Figure 4.15: ROC Curves for Sequential Full Model in Confirmation Set

4.4.1.2 Sequential, Optimal

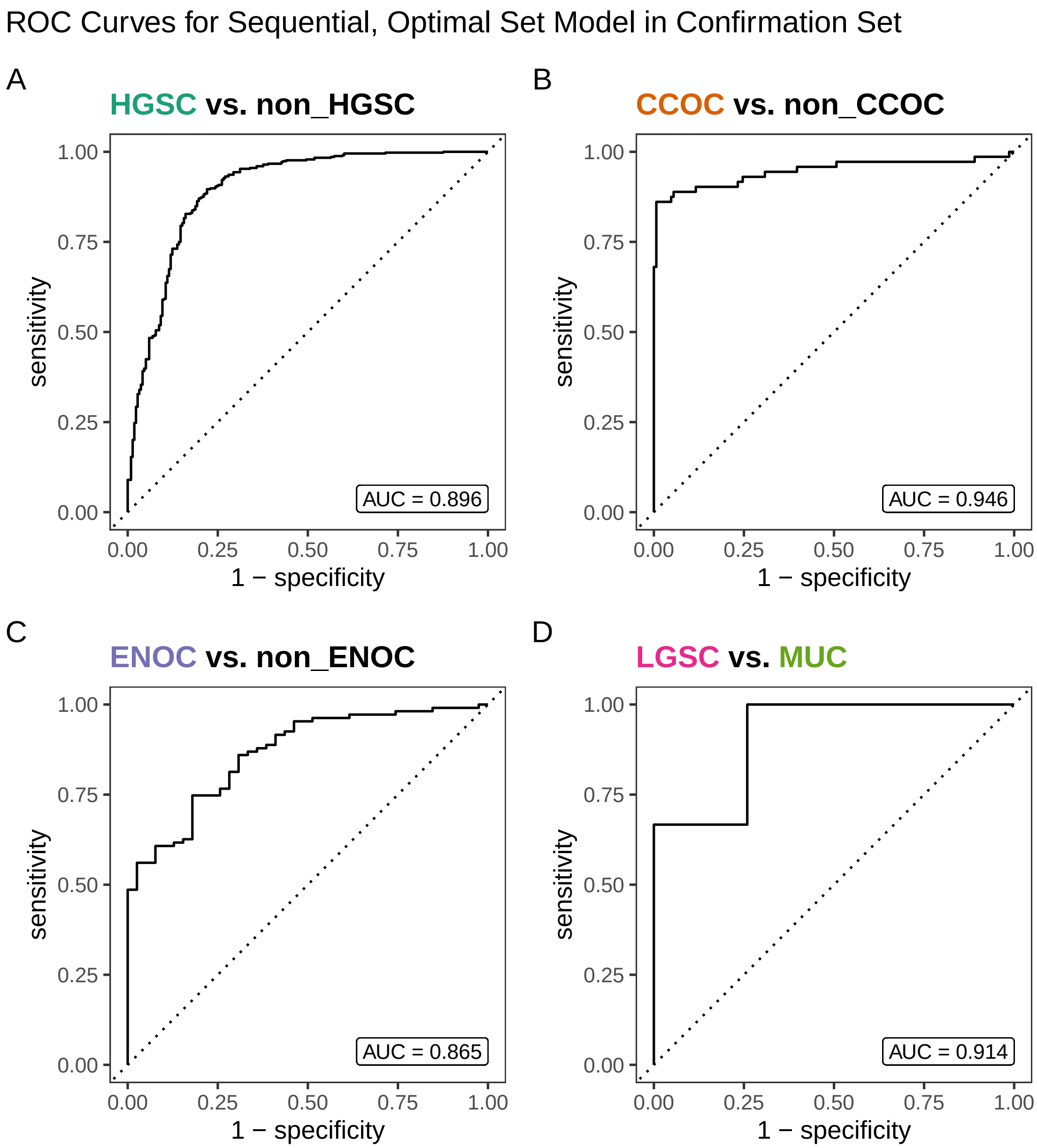

Figure 4.16: ROC Curves for Sequential, Optimal Model in Confirmation Set

4.4.1.3 SMOTE-Random Forest, Full

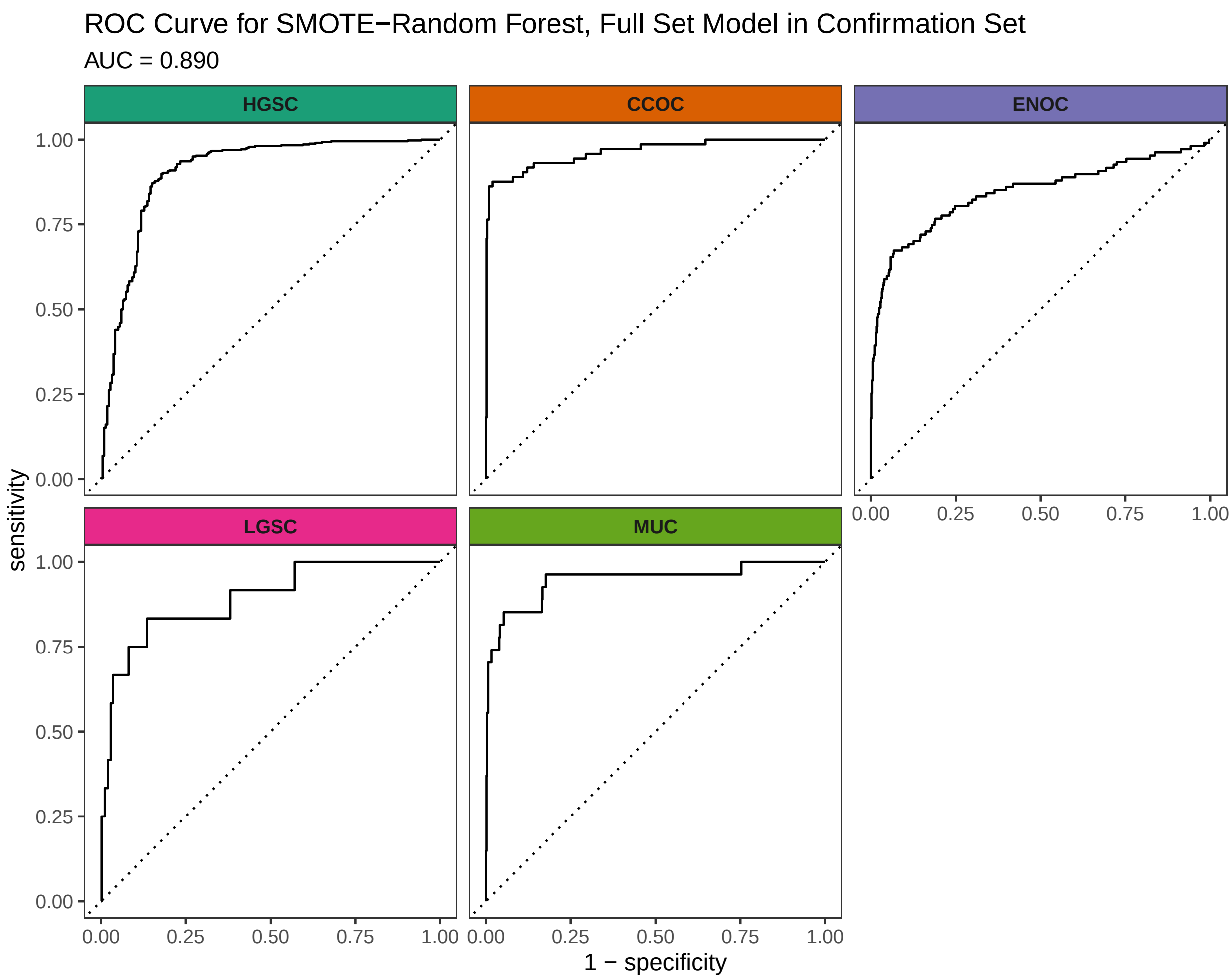

Figure 4.17: ROC Curves for SMOTE-Random Forest, Full Set Model in Confirmation Set

4.4.1.4 SMOTE-Random Forest, Optimal

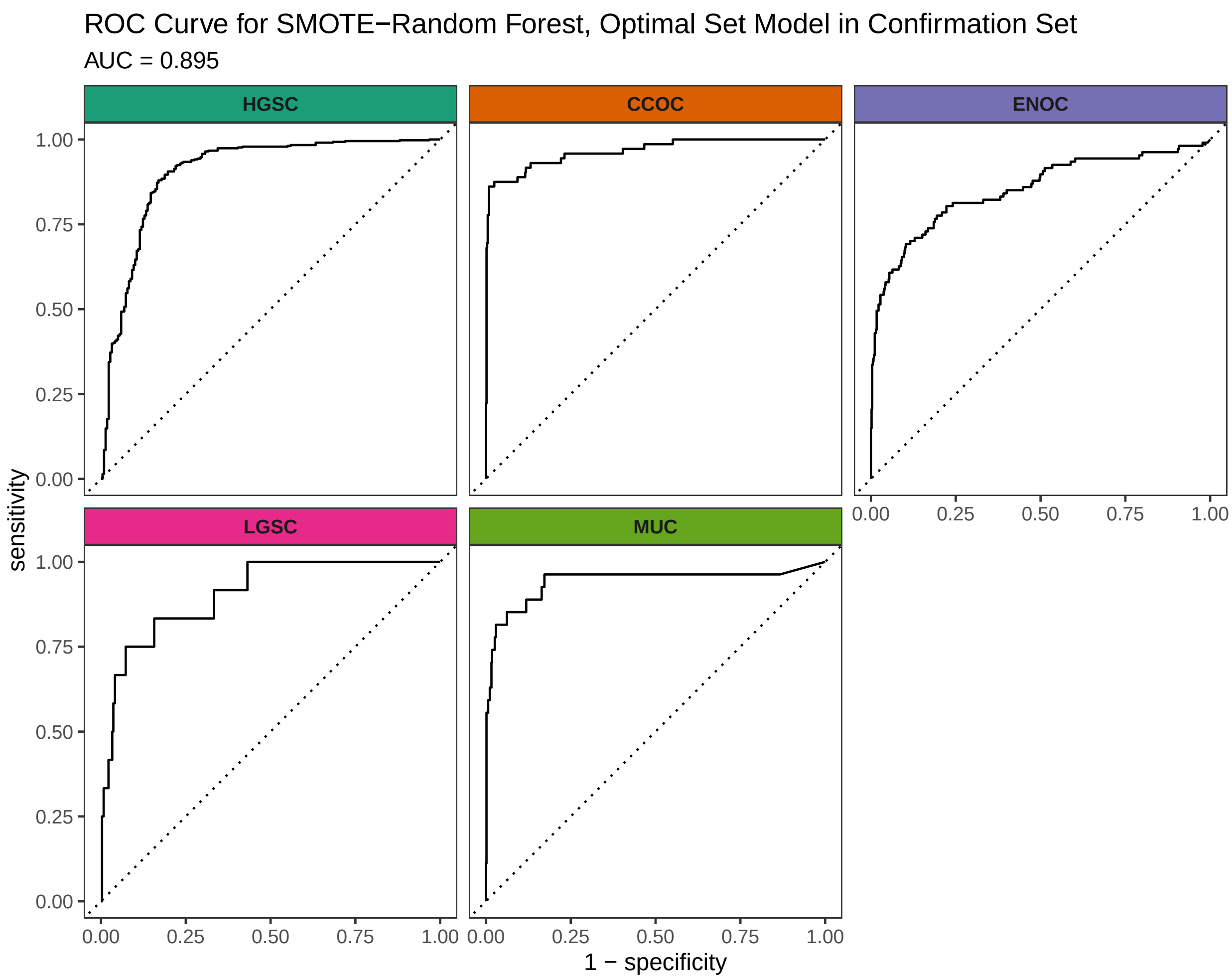

Figure 4.18: ROC Curves for SMOTE-Random Forest, Optimal Set Model in Confirmation Set

4.4.1.5 Two-Step, Full

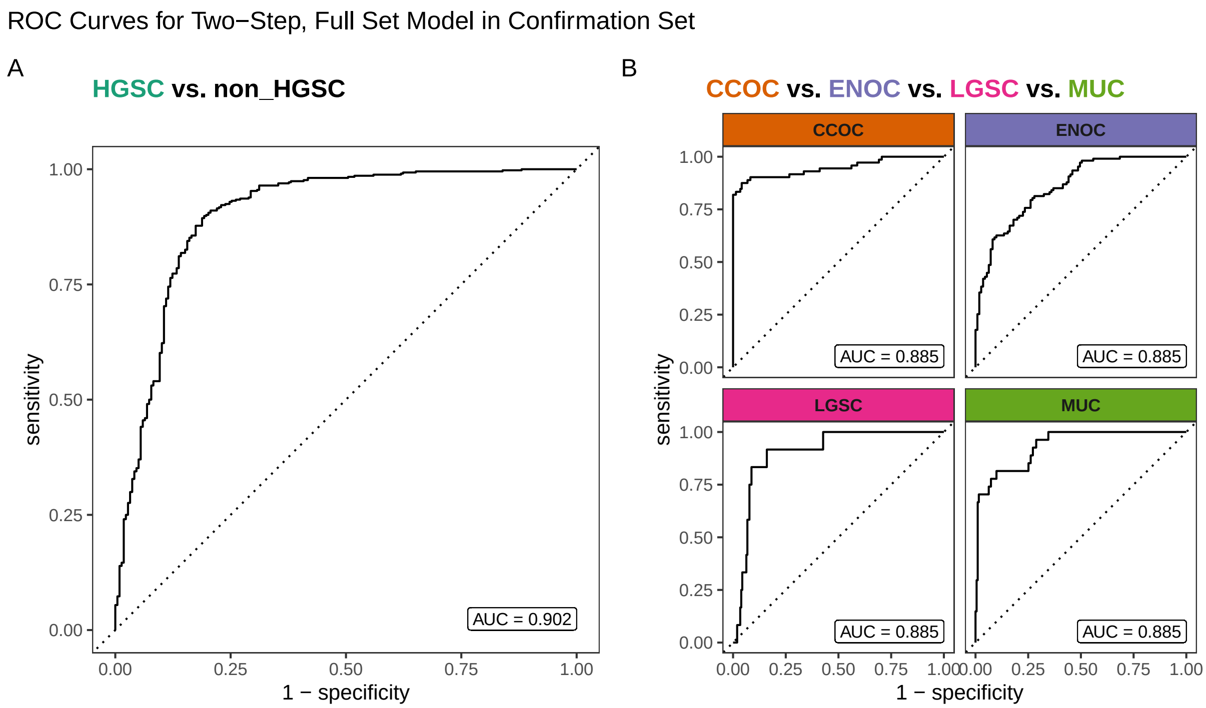

Figure 4.19: ROC Curves for Two-Step Full Model in Confirmation Set

4.4.1.6 Two-Step, Optimal

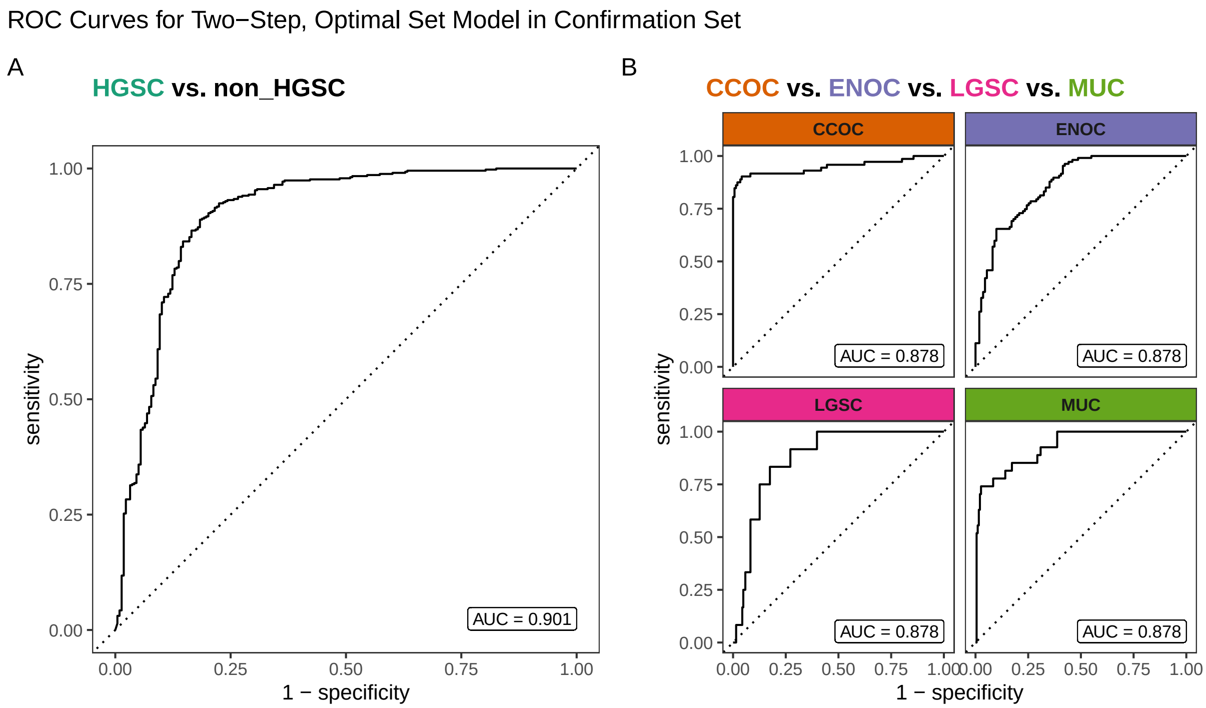

Figure 4.20: ROC Curves for Two-Step Optimal Model in Confirmation Set

4.4.2 Validation Set

Table 4.18: Evaluation Metrics on Validation Set Model, SMOTE-Random Forest, Optimal Set

		Histotypes				
Metric	Overall	HGSC	CCOC	ENOC	LGSC	MUC
Accuracy	0.889	0.907	0.970	0.946	0.977	0.979
Sensitivity	0.781	0.917	0.971	0.682	0.467	0.870
Specificity	0.957	0.872	0.970	0.975	0.985	0.982
F1-Score	0.713	0.939	0.832	0.714	0.400	0.678
Balanced Accuracy	0.869	0.894	0.970	0.829	0.726	0.926
Kappa	0.723	0.743	0.816	0.685	0.388	0.668

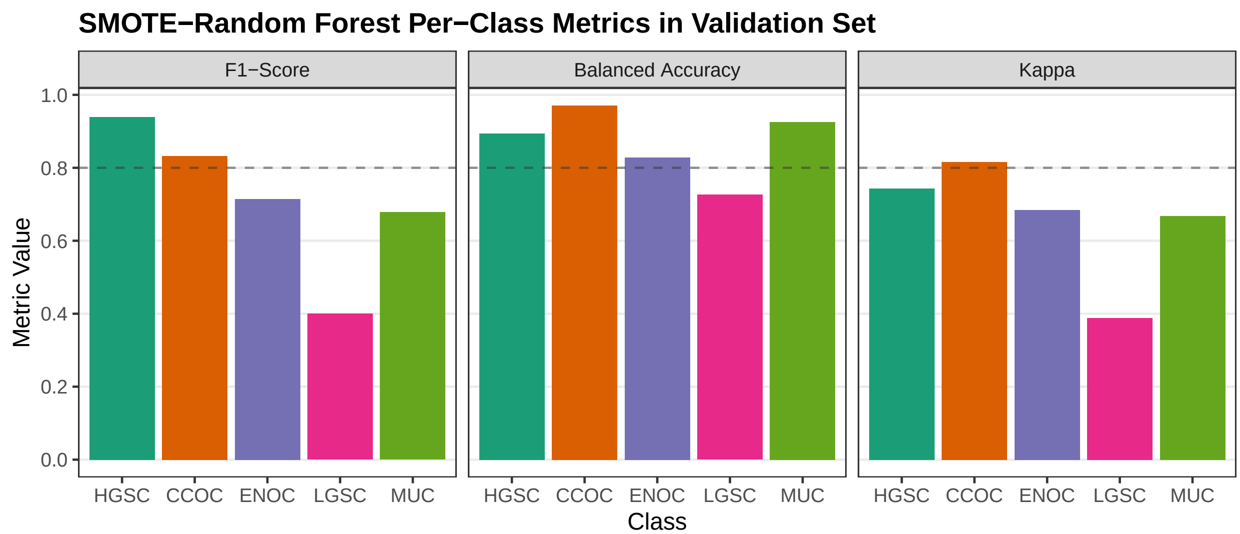

Figure 4.21: SMOTE-Random Forest Per-Class Metrics in Validation Set

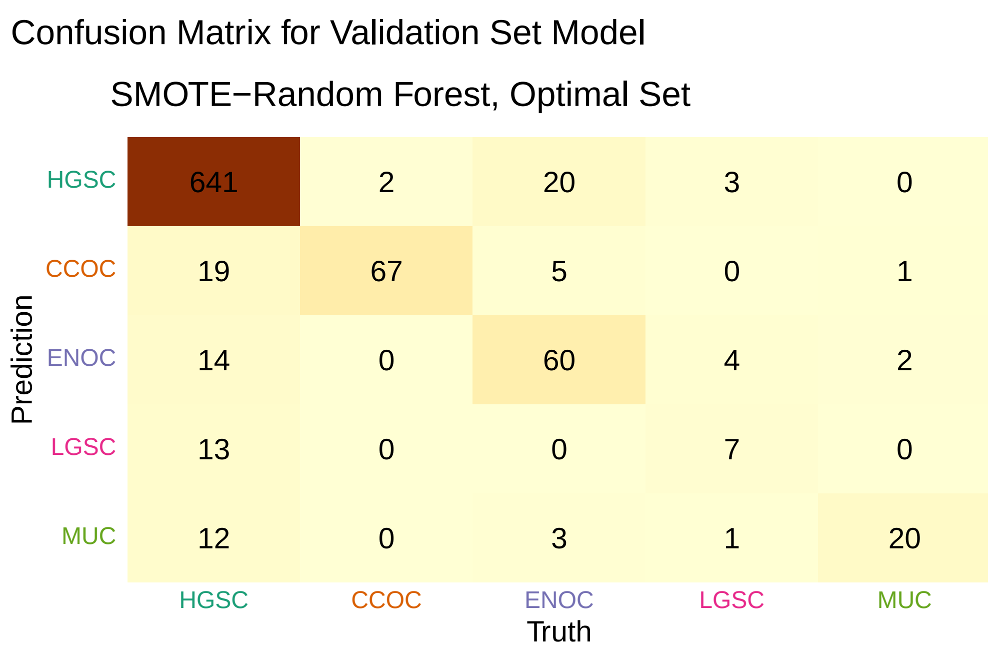

Figure 4.22: Confusion Matrix for Validation Set Model

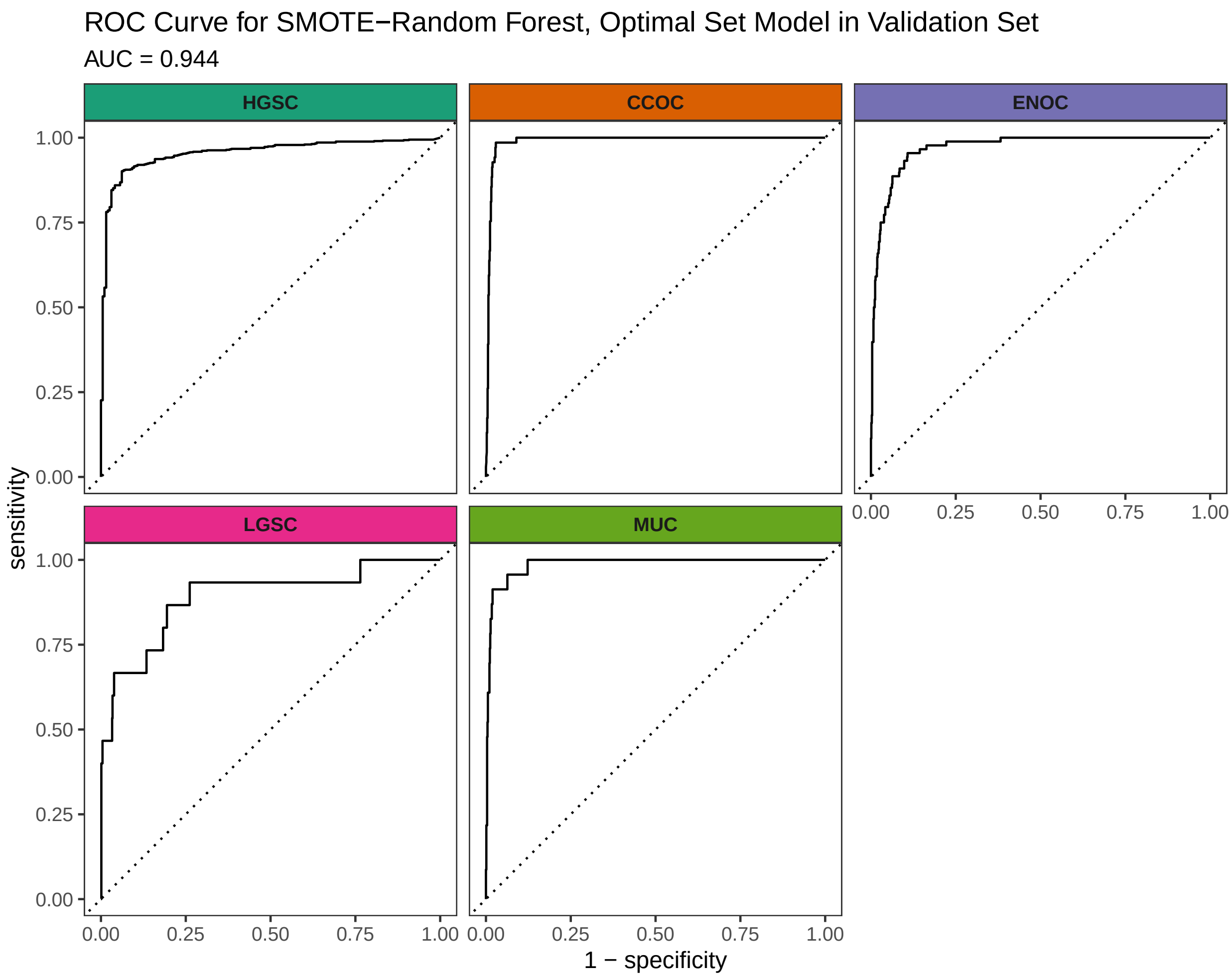

Figure 4.23: ROC Curves for SMOTE-Random Forest, Optimal Set Model in Validation Set

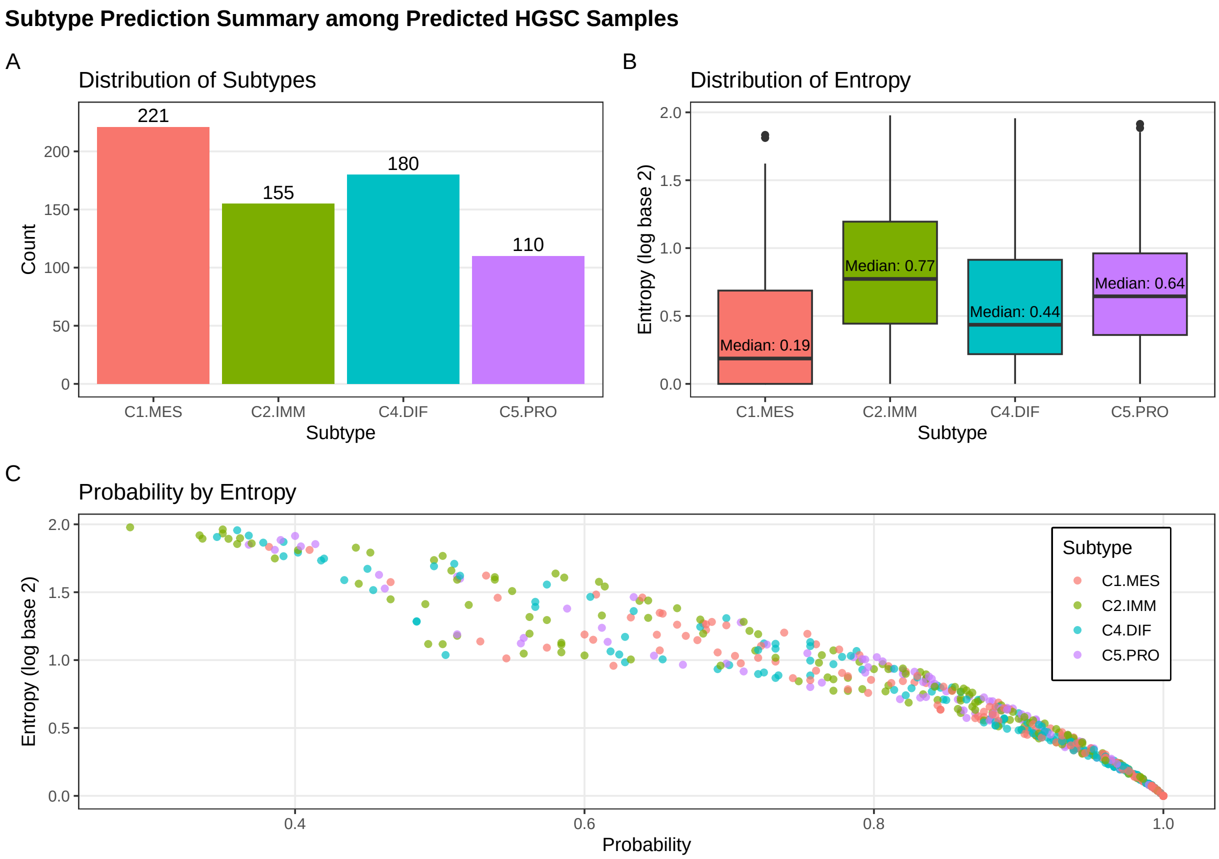

Figure 4.24: Subtype Prediction Summary among Predicted HGSC Samples

References

Talhouk, Aline, Stefan Kommoss, Robertson Mackenzie, Martin Cheung, Samuel Leung, Derek S. Chiu, Steve E. Kalloger, et al. 2016. "Single-Patient Molecular Testing with NanoString nCounter Data Using a Reference-Based Strategy for Batch Effect Correction." Edited by Benjamin Haibe-Kains. *PLOS ONE* 11 (4): e0153844. https://doi.org/10.1371/journal.pone.0153844.