Ovarian Cancer Histotypes: Report of Statistical Findings

Derek Chiu

2024-03-19

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, Kappa, and G-mean to inform the decision of which methods to carry forward for prediction in confirmation and validation sets.

2. Methods

2.1 Normalization

The full training set was comprised of data from CodeSet (CS) 1, 2, and 3. All CodeSets were first normalized to housekeeping genes, then a different approach was taken for each of the CodeSets.

CS1 was normalized to CS3 using "Random1" reference samples. These reference samples are common samples between CS1 and CS3, randomly selected such that we obtain one from each of the five histotypes. Then we use the reference method to normalize CS1 to CS3.

Similarly, CS2 was normalized to CS3 using "Random1" reference samples using five common samples between CS2 and CS3 such that there is one from each histotype.

For CS3, we first split the dataset by site: Vancouver, USC, and AOC. We use the CS3-Vancouver subset as a "reference standard", so we normalized CS3-USC and CS3-AOC to CS3-Vancouver using a "Random1" reference method where we reference samples are common between USC and Vancouver, and between AOC and Vancouver. The CS3-Vancouver is also included without further normalization.

2.2 Case Selection

Duplicate cases (two samples with the same ottaID) were removed from the training set before fitting the classification models. CS3 cases were preferred over CS1 and CS2, and CS3-Vancouver were preferred over CS3-AOC and CS3-USC.

The training, confirmation, and validation sets all used a different set of cohorts.

2.3 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`)

2.3.1 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.

2.3.2 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 within ranges (transformed scale) [0, 2] and [-3, 0], respectively.
- `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.

2.3.3 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:

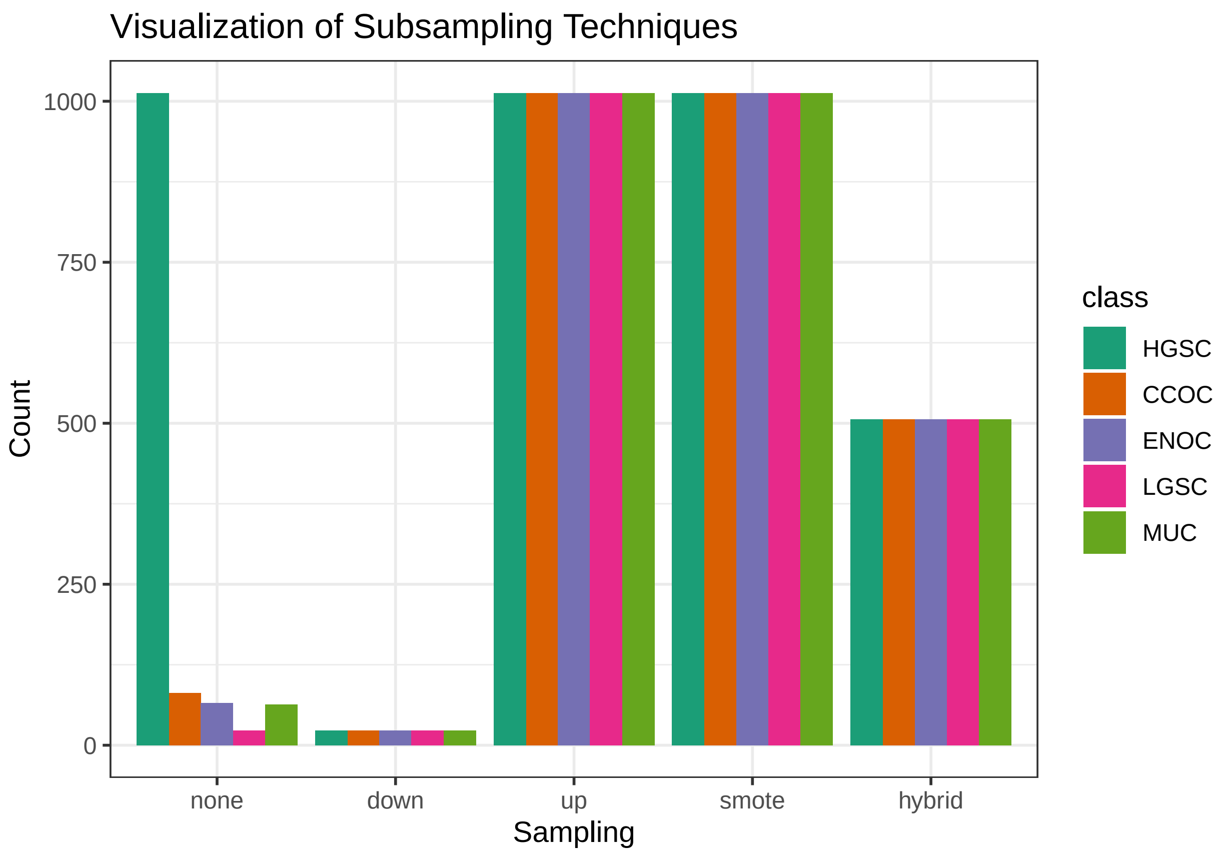


Figure 2.1: Visualization of Subsampling Techniques

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).

Let

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

Figure 2.2 illustrates how the sequential algorithm works for K=5, using ovarian histotypes as an example for the classes.

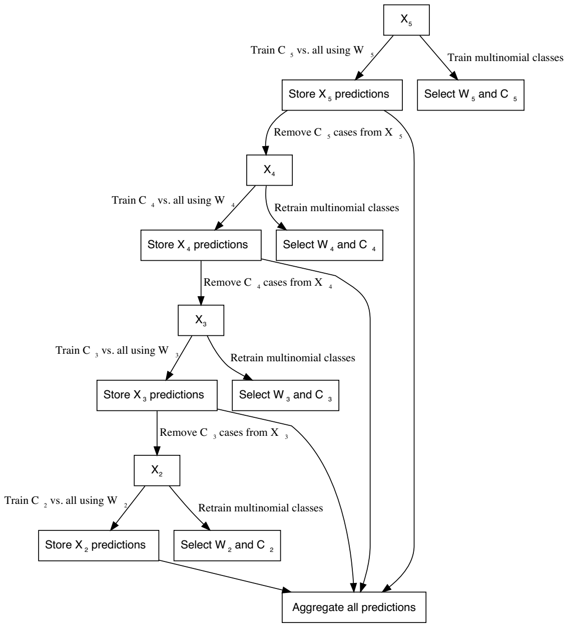

X₅
Train C 5 vs. all using W 5
Train multinomial classes
Store X 5 predictions
Select W 5 and C 5
Remove C 5 cases from X 5
X₄
Train C 4 vs. all using W 4
Retrain multinomial classes
Store X 4 predictions
Select W 4 and C 4
Remove C 4 cases from X 4
X₃
Train C 3 vs. all using W 3
Retrain multinomial classes
Store X 3 predictions
Select W 3 and C 3
Remove C 3 cases from X 3
X₂
Train C 2 vs. all using W 2
Retrain multinomial classes
Store X 2 predictions
Select W 2 and C 2
Aggregate all predictions

Figure 2.2: Sequential Algorithm

2.4.1 Subsampling

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.

2.5 Two-Step Algorithm

The two-step algorithm can be thought of as a special case of the sequential algorithm, that is specific to classifying ovarian histotypes. 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. Thus, we can implement a two-step algorithm as such:

- Step 1: use binary classification for HGSC vs. non-HGSC (this step is the same as step 1 in the sequential algorithm above)
- Step 2: use multinomial classification for remaining non-HGSC classes

Using some of the notation from Equation (2.4), a flowchart similar to Figure 2.2 can show how the two-step algorithm works:

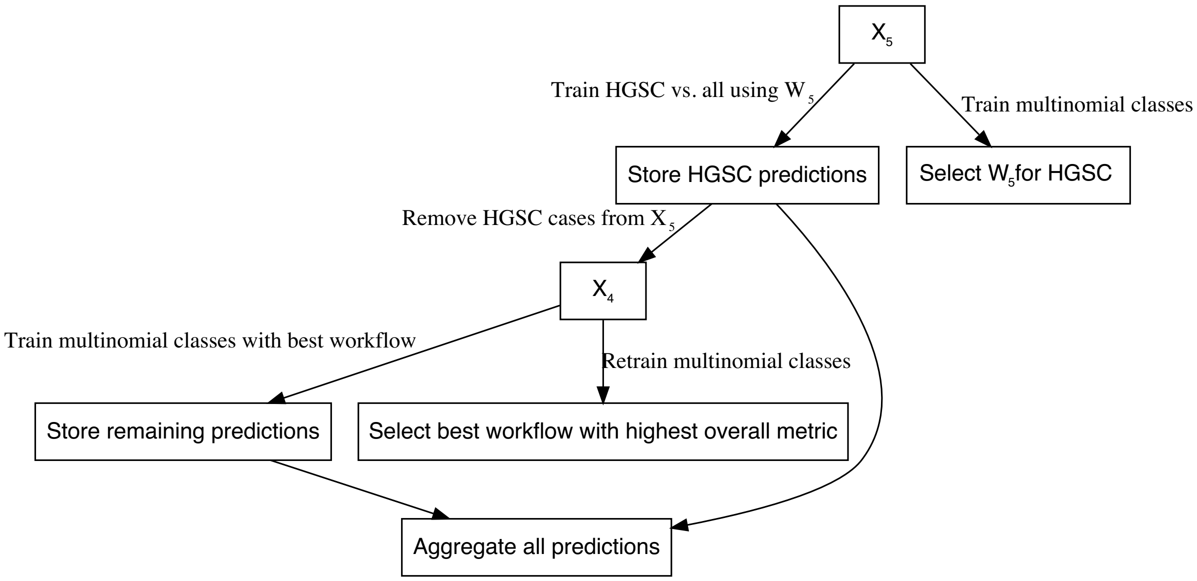

X5
Train HGSC vs. all using W5
Train multinomial classes
Store HGSC predictions
Select W5for HGSC
Remove HGSC cases from X5
X4
Train multinomial classes with best workflow
Retrain multinomial classes
Store remaining predictions
Select best workflow with highest overall metric
Aggregate all predictions

Figure 2.3: Two-Step Algorithm

3. Distributions

3.1 Histotypes in Classifier Data

3.2 Cohort Counts

3.3 Cohorts in Classifier Data

3.4 Quality Control

3.4.1 Failed Samples

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

- `linFlag`: linearity of positive controls with positive control concentrations is less than 0.95, or linearity measures are unknown
- `imagingFlag`: percent of field of view is less than 75%
- `spcFlag`: smallest positive control is less than the lower limit of detection (negative control average expression less two times the negative control standard deviation), or negative control average expression equals zero
- `normFlag`: signal to noise ratio less than 100, or percent of genes detected is less than 50. Note: these thresholds were determined by examining the %GD vs. SNR relationship below.

3.4.2 %GD vs. SNR

\begin{figure}[H]

Table 3.1: Pre-QC Training Set Histotype Distribution by CodeSet

Variable	Levels	CS1	CS2	CS3	Total
Histotype	HGSC	120 (45%)	643 (79%)	515 (92%)	1278 (78%)
	CCOC	48 (18%)	61 (7%)	11 (2%)	120 (7%)
	ENOC	60 (22%)	32 (4%)	11 (2%)	103 (6%)
	MUC	19 (7%)	62 (8%)	12 (2%)	93 (6%)
	LGSC	20 (7%)	21 (3%)	9 (2%)	50 (3%)
Total	N (%)	267 (16%)	819 (50%)	558 (34%)	1644 (100%)

Table 3.2: Training Set (with duplicates) Histotype Distribution by CodeSet

Variable	Levels	CS1	CS2	CS3	Total
Histotype	HGSC	116 (48%)	623 (80%)	475 (94%)	1214 (79%)
	CCOC	44 (18%)	54 (7%)	8 (2%)	106 (7%)
	ENOC	55 (23%)	27 (3%)	8 (2%)	90 (6%)
	MUC	15 (6%)	59 (8%)	9 (2%)	83 (5%)
	LGSC	14 (6%)	19 (2%)	6 (1%)	39 (3%)
Total	N (%)	244 (16%)	782 (51%)	506 (33%)	1532 (100%)

Table 3.3: Final Training Set Histotype Distribution by CodeSet

Variable	Levels	CS1	CS2	CS3	Total
Histotype	HGSC	9 (12%)	553 (79%)	451 (96%)	1013 (81%)
	CCOC	25 (32%)	52 (7%)	4 (1%)	81 (7%)
	ENOC	37 (48%)	25 (4%)	4 (1%)	66 (5%)
	MUC	3 (4%)	55 (8%)	5 (1%)	63 (5%)
	LGSC	3 (4%)	16 (2%)	4 (1%)	23 (2%)
Total	N (%)	77 (6%)	701 (56%)	468 (38%)	1246 (100%)

Table 3.4: Histotype Distribution in Confirmation and Validation Sets

Variable	Levels	Confirmation	Validation
Histotype	HGSC	422 (66%)	674 (74%)
	CCOC	75 (12%)	80 (9%)
	ENOC	106 (16%)	108 (12%)
	MUC	27 (4%)	26 (3%)
	LGSC	13 (2%)	18 (2%)
Total	N (%)	643 (42%)	906 (58%)

Table 3.5: Training Set counts by CodeSet and Processing Stage

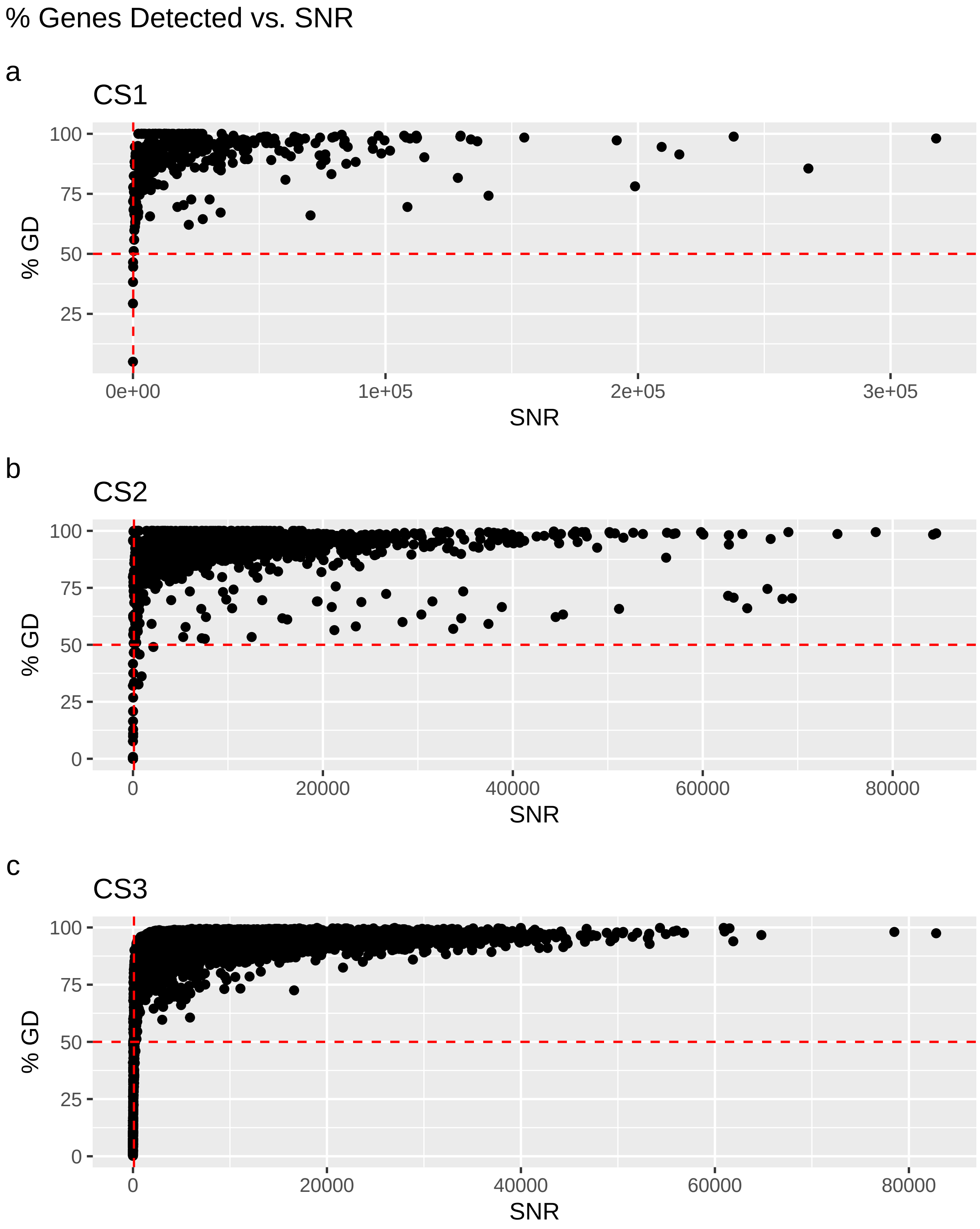
Processing Stage	CS1	CS2	CS3	Total
Raw Data	412	1223	5424	7059
Selected Cohorts	294	903	2477	3674
QC	286	888	2285	3459
Normalized to Reference	263	832	2107	3202
CS3: remove test sets, add AOC/USC	263	832	514	1609
Major Histotypes	244	782	506	1532
Removed Duplicates	77	701	468	1246

Table 3.6: Cohort Distribution in Training, Confirmation, and Validation Sets

CodeSet	Cohort	Training	Confirmation	Validation
CS1	MAYO	2	0	0
CS1	MTL	1	0	0
CS1	OOU	53	0	0
CS1	OOUE	1	0	0
CS1	VOA	20	0	0
CS2	ICON7	365	0	0
CS2	JAPAN	8	0	0
CS2	MAYO	42	0	0
CS2	MTL	59	0	0
CS2	OOU	27	0	0
CS2	OOUE	18	0	0
CS2	OVAR3	136	0	0
CS2	VOA	46	0	0
CS3	OOU	18	0	0
CS3	OOUE	11	0	0
CS3	VOA	439	0	0
CS3	TNCO	0	643	0
CS3	DOVE4	0	0	906

Table 3.7: Number of failed samples by CodeSet and fail condition

CodeSet	CodeSet Total	linFlag	imagingFlag	spcFlag	normFlag	QCFlag	n
CS1	8	Passed	Failed	Passed	Passed	Failed	3
		Passed	Passed	Passed	Failed	Failed	5
CS2	32	Failed	Failed	Failed	Failed	Failed	2
		Failed	Passed	Failed	Failed	Failed	3
		Failed	Passed	Passed	Passed	Failed	3
		Passed	Failed	Passed	Passed	Failed	3
		Passed	Passed	Passed	Failed	Failed	21
CS3	274	Failed	Failed	Failed	Failed	Failed	1
		Failed	Failed	Passed	Failed	Failed	3
		Failed	Passed	Passed	Failed	Failed	11
		Passed	Failed	Passed	Passed	Failed	7
		Passed	Passed	Passed	Failed	Failed	252

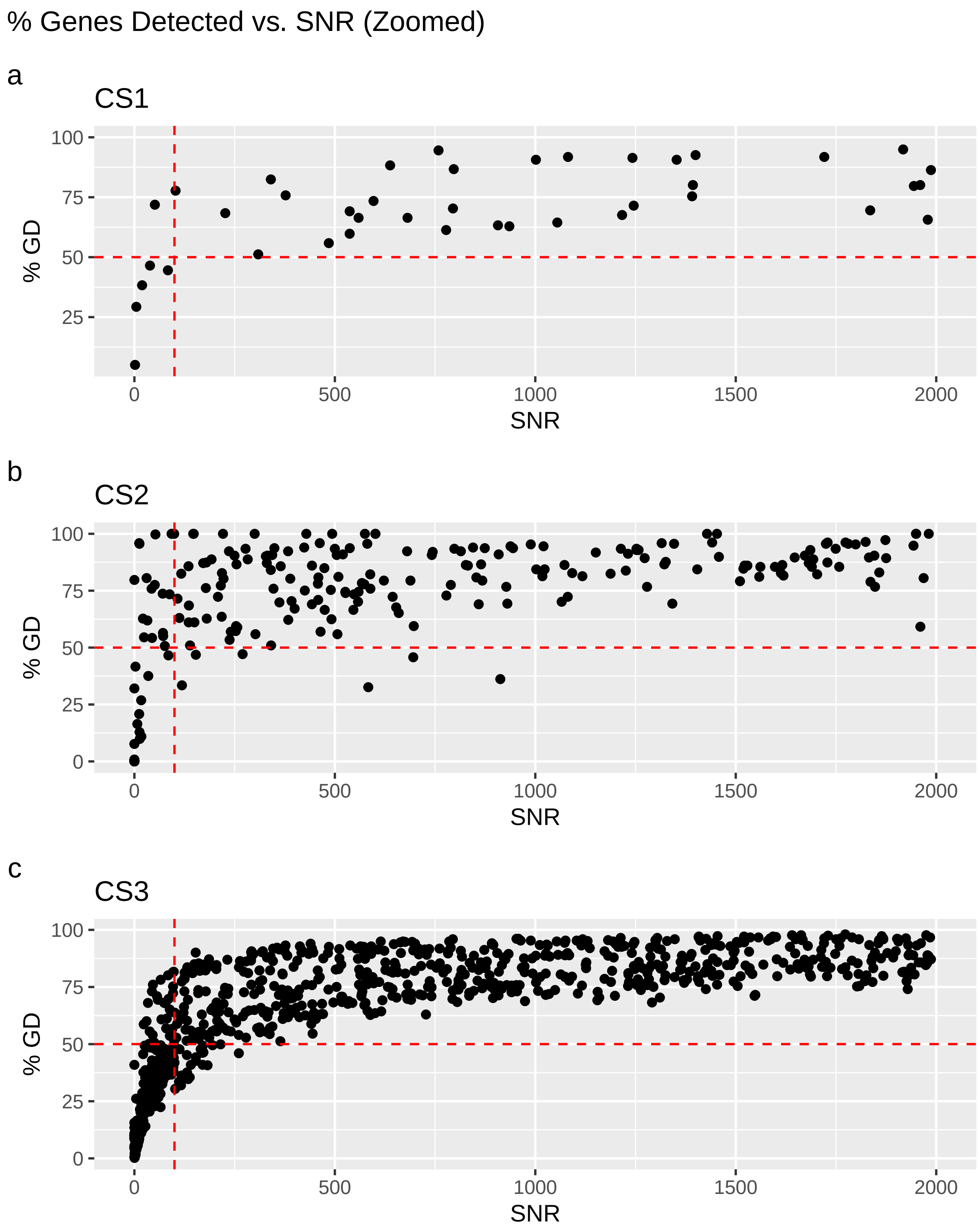


{

}

\caption{% Genes Detected vs. Signal to Noise Ratio} \end{figure}

\begin{figure}[H]



{

}

\caption{% Genes Detected vs. Signal to Noise Ratio (Zoomed)} \end{figure}

3.5 Pairwise Gene Expression

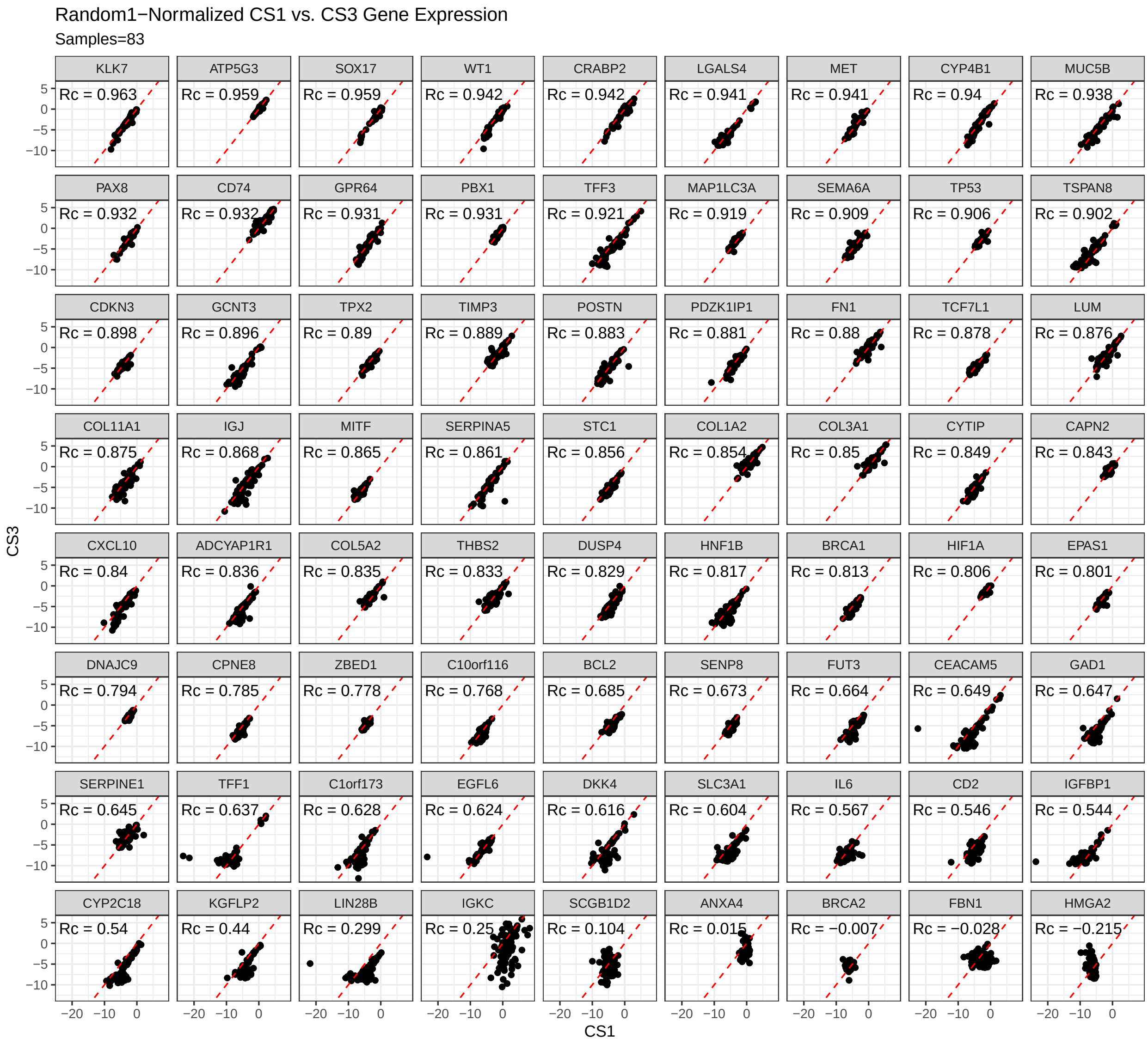


Figure 3.1: Random1-Normalized CS1 vs. CS3 Gene Expression

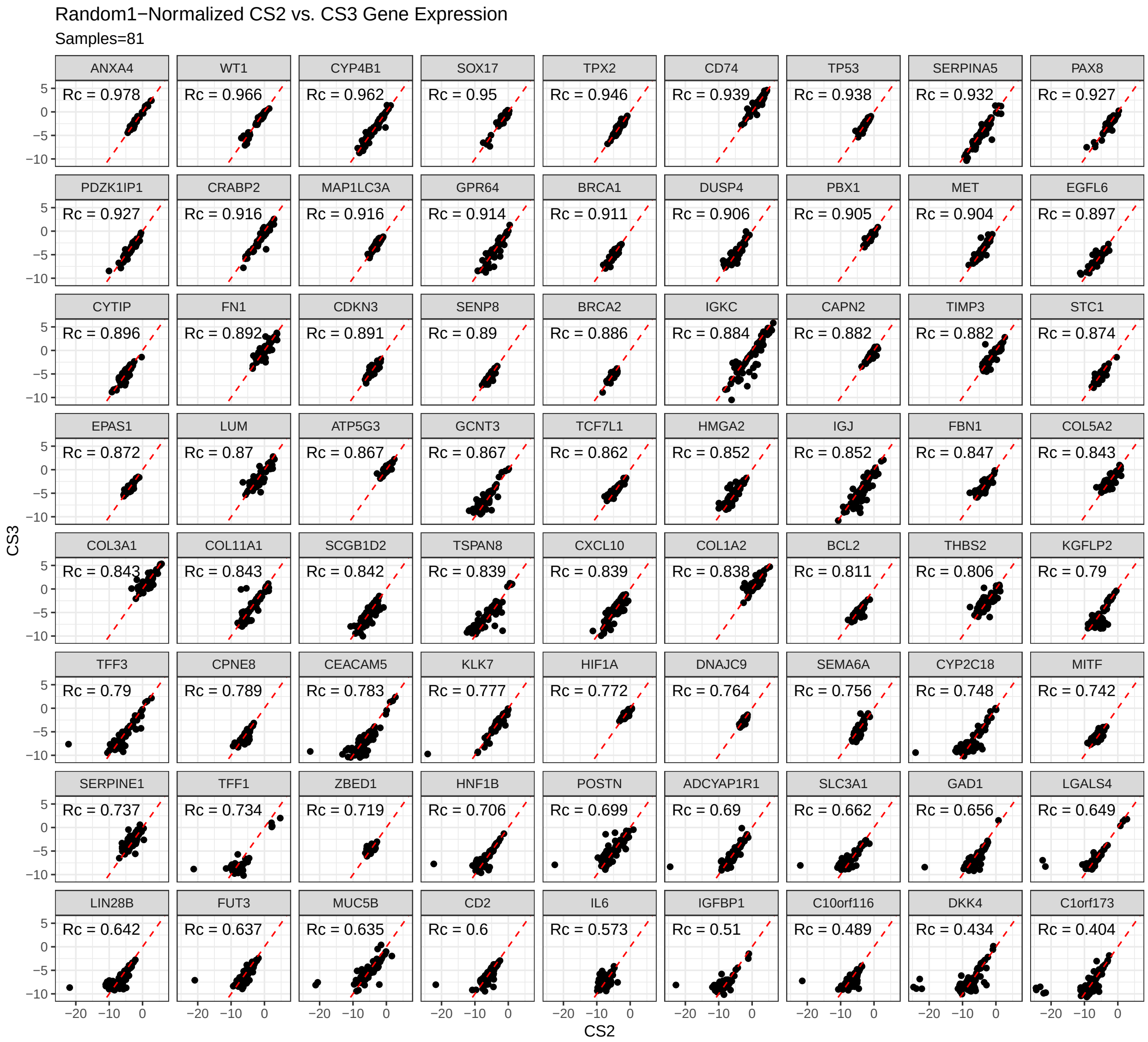


Figure 3.2: Random1-Normalized CS2 vs. CS3 Gene Expression

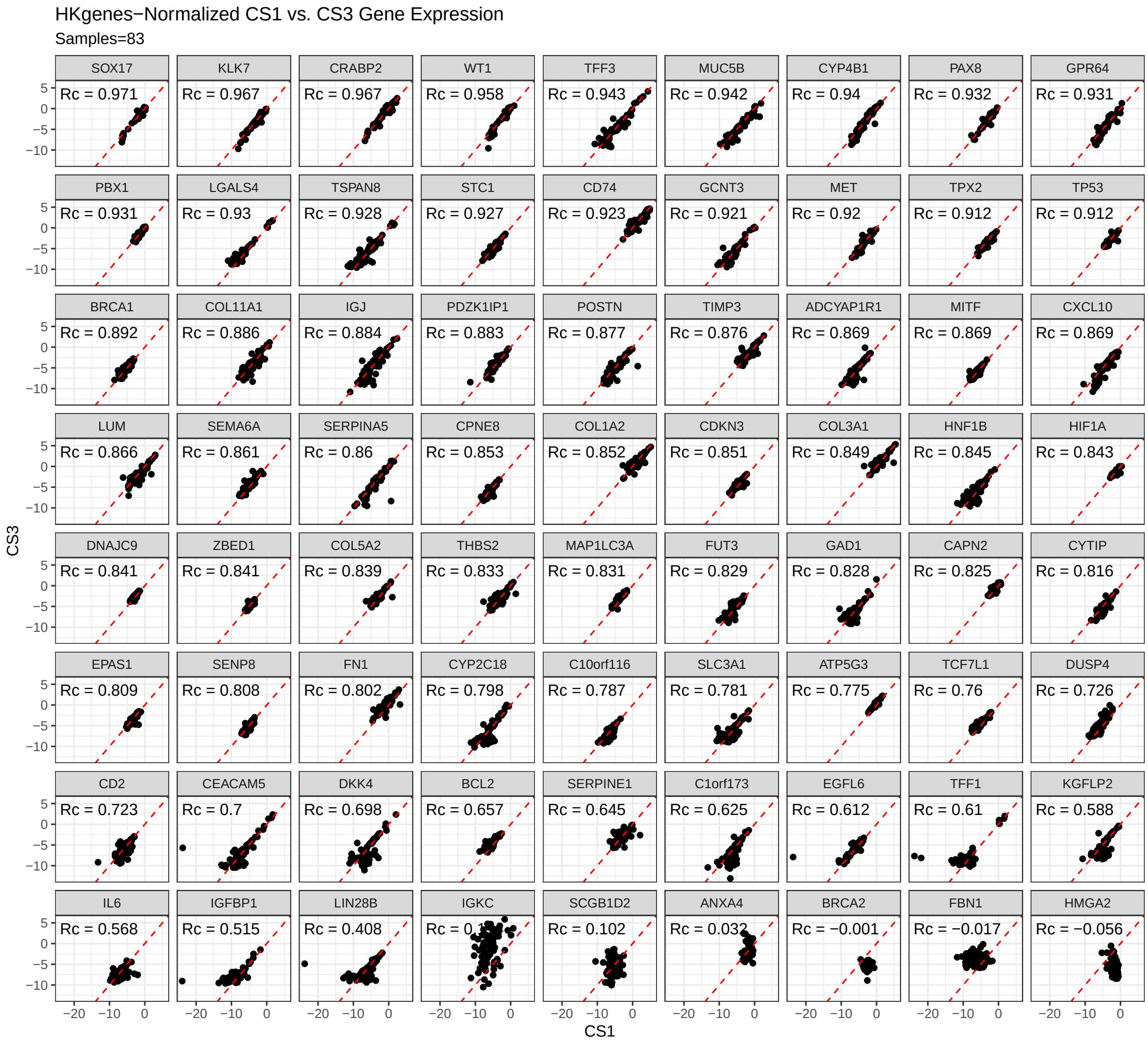


Figure 3.3: HKgenes-Normalized CS1 vs. CS3 Gene Expression

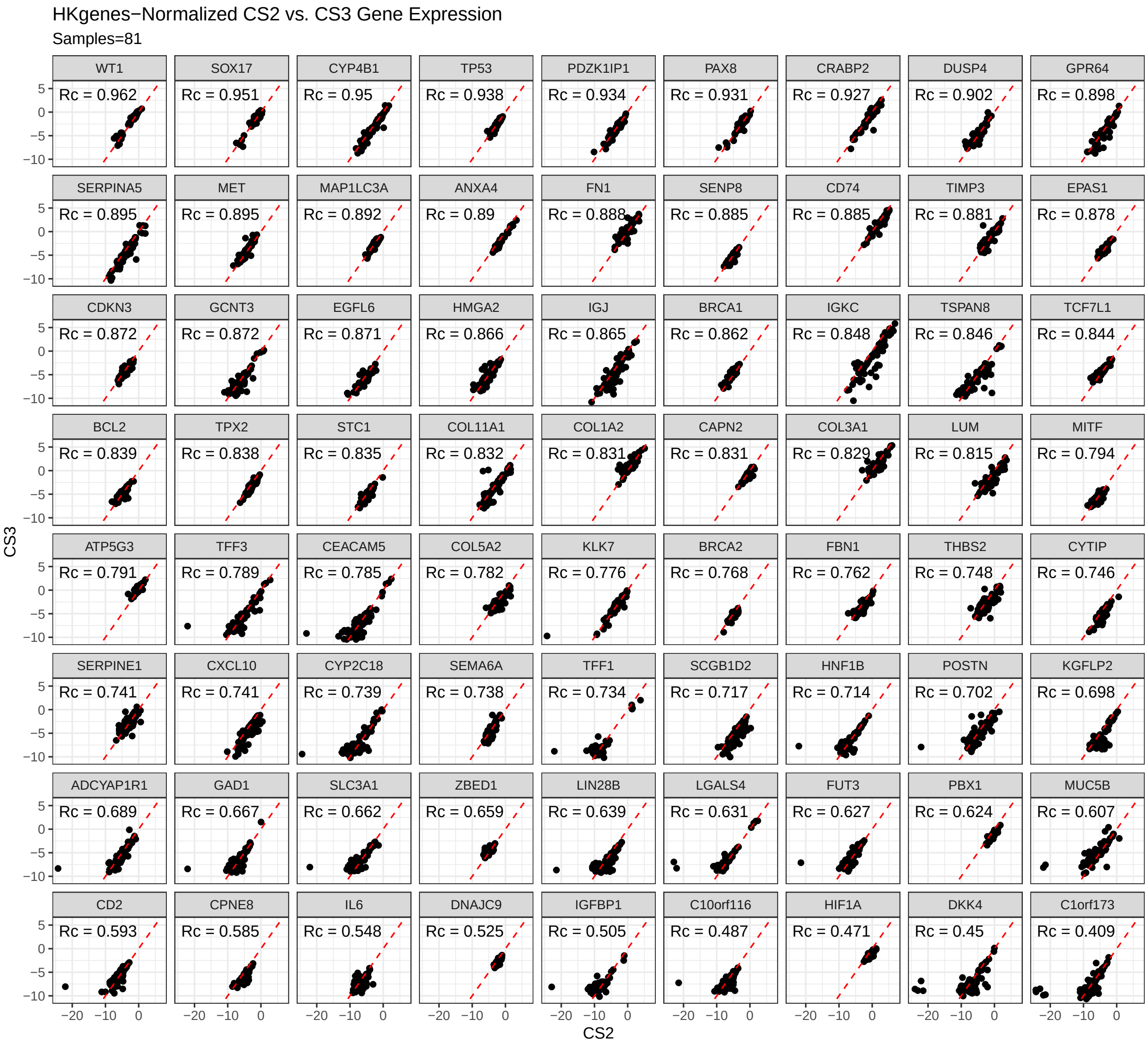


Figure 3.4: HKgenes-Normalized CS2 vs. CS3 Gene Expression

4. Results

We show internal validation summaries for the combined classifier training set, as well as the CS1 and CS2 sets with duplicates included. The F1-scores, kappa, and G-mean are the measures of interest. Algorithms are sorted by descending value based on the overallaccuracy of the training set. The point ranges show the median, 5th and 95th percentiles, coloured by subsampling methods.

4.1 Training Set

4.1.1 Accuracy

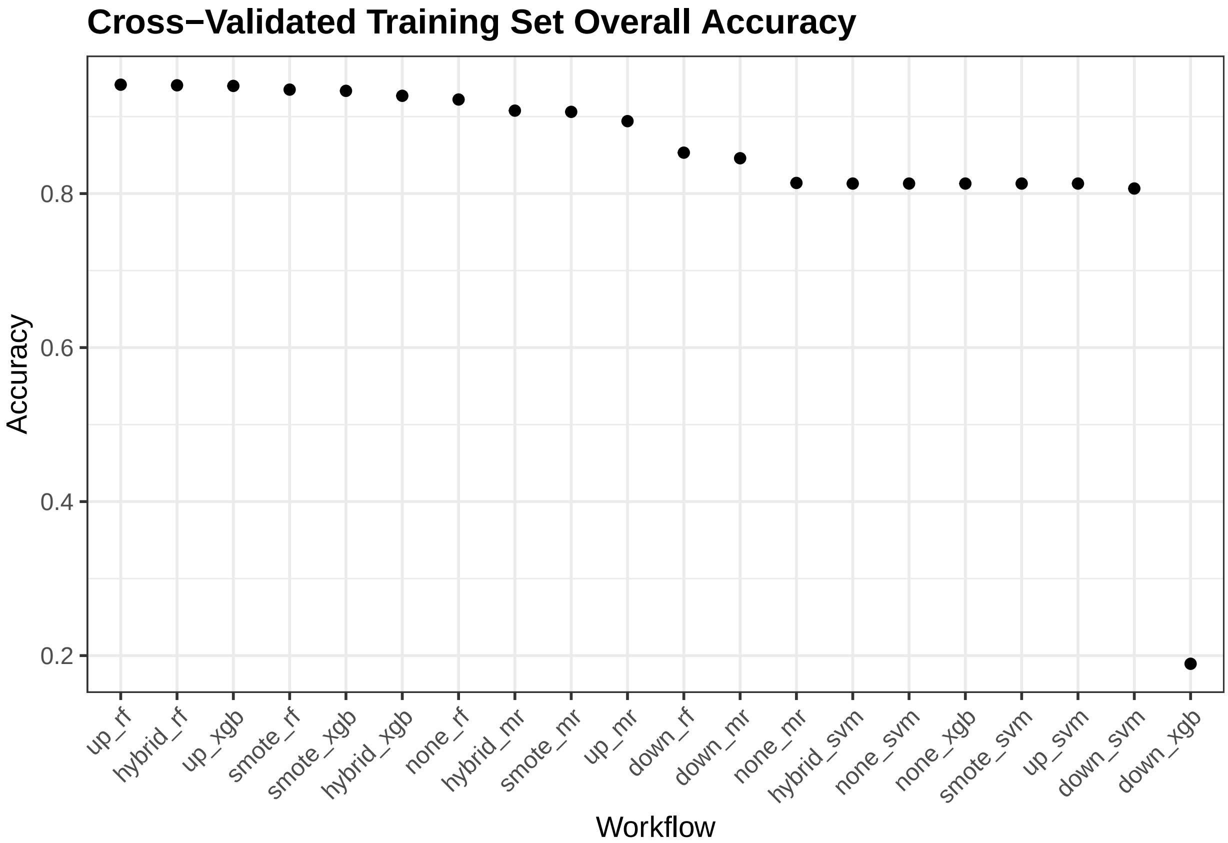


Figure 4.1: Training Set Accuracy

Table 4.1: Cross-Validated Training Set Overall Accuracy

samp	mr	rf	svm	xgb
none	0.814	0.922	0.813	0.813
down	0.846	0.853	0.807	0.189
up	0.894	**0.941**	0.813	0.94
smote	0.906	0.935	0.813	0.933
hybrid	0.908	**0.941**	0.813	0.927

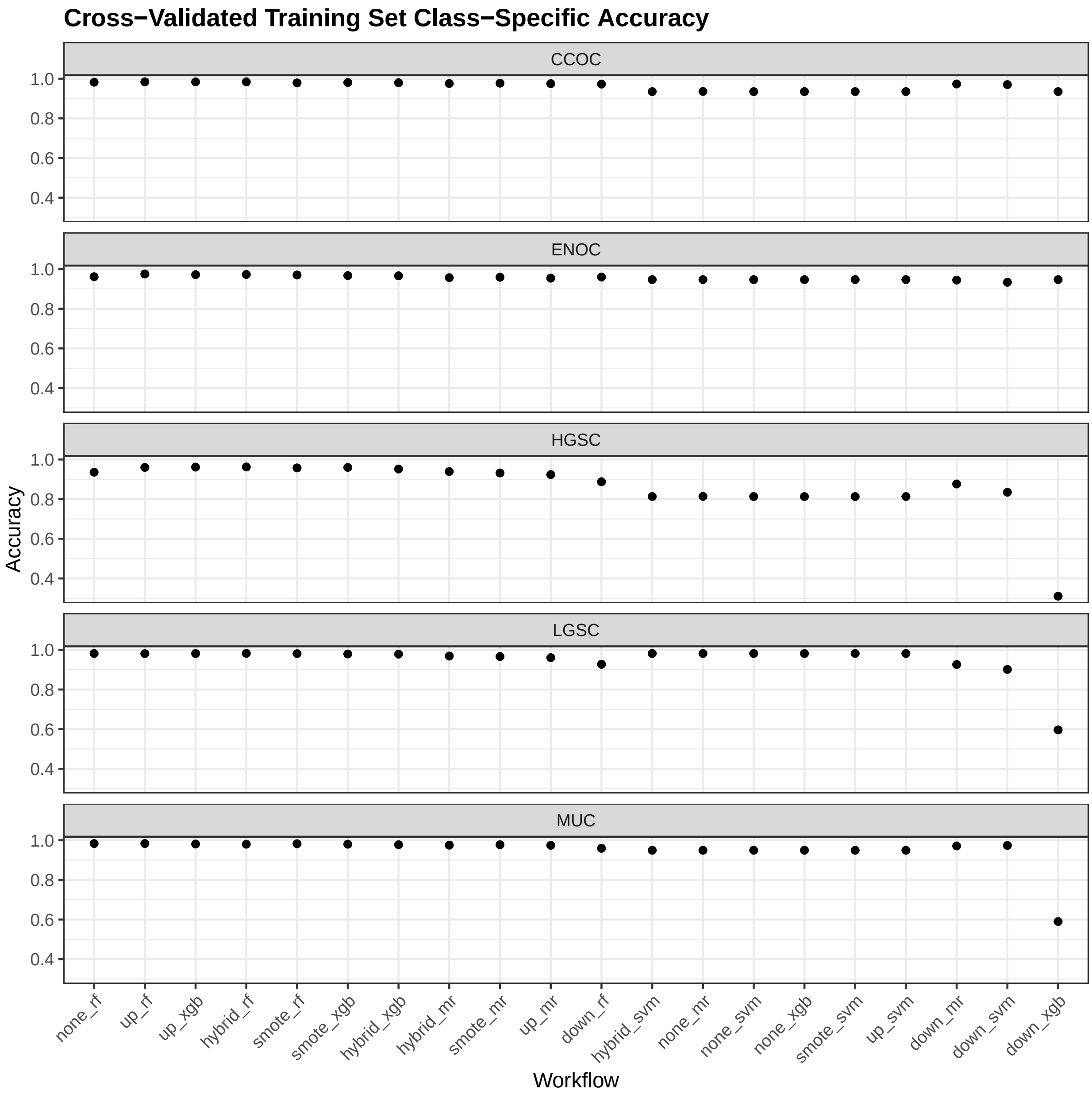


Figure 4.2: Training Set Class-Specific Accuracy

Table 4.2: Cross-Validated Training Set Class-Specific Accuracy

samp	histotype	mr	rf	svm	xgb
none	CCOC	0.936	0.982	0.935	0.935
none	ENOC	0.947	0.961	0.947	0.947
none	HGSC	0.814	0.936	0.813	0.813
none	LGSC	0.982	0.982	0.982	0.982
none	MUC	0.949	0.983	0.949	0.949
down	CCOC	0.974	0.973	0.97	0.935
down	ENOC	0.945	0.96	0.933	0.947
down	HGSC	0.876	0.888	0.835	0.311
down	LGSC	0.926	0.927	0.901	0.596
down	MUC	0.971	0.959	0.974	0.59
up	CCOC	0.975	**0.984**	0.935	**0.984**
up	ENOC	0.954	0.975	0.947	0.972
up	HGSC	0.924	0.96	0.813	0.961
up	LGSC	0.961	0.981	0.982	0.982
up	MUC	0.974	0.983	0.949	0.981
smote	CCOC	0.978	0.979	0.935	0.981
smote	ENOC	0.959	0.97	0.947	0.967
smote	HGSC	0.932	0.957	0.813	0.96
smote	LGSC	0.966	0.981	0.982	0.979
smote	MUC	0.978	0.982	0.949	0.98
hybrid	CCOC	0.976	**0.984**	0.935	0.98
hybrid	ENOC	0.957	0.973	0.947	0.966
hybrid	HGSC	0.939	0.962	0.813	0.952
hybrid	LGSC	0.969	0.982	0.982	0.978
hybrid	MUC	0.975	0.98	0.949	0.978

Table 4.3: Cross-Validated Training Set Overall F1-Score

samp	mr	rf	svm	xgb
none	0.461	0.795	**0.897**	**0.897**
down	0.672	0.665	0.648	0.148
up	0.717	0.735	**0.897**	0.756
smote	0.74	0.737	**0.897**	0.752
hybrid	0.73	0.773	**0.897**	0.748

4.1.2 F1-Score

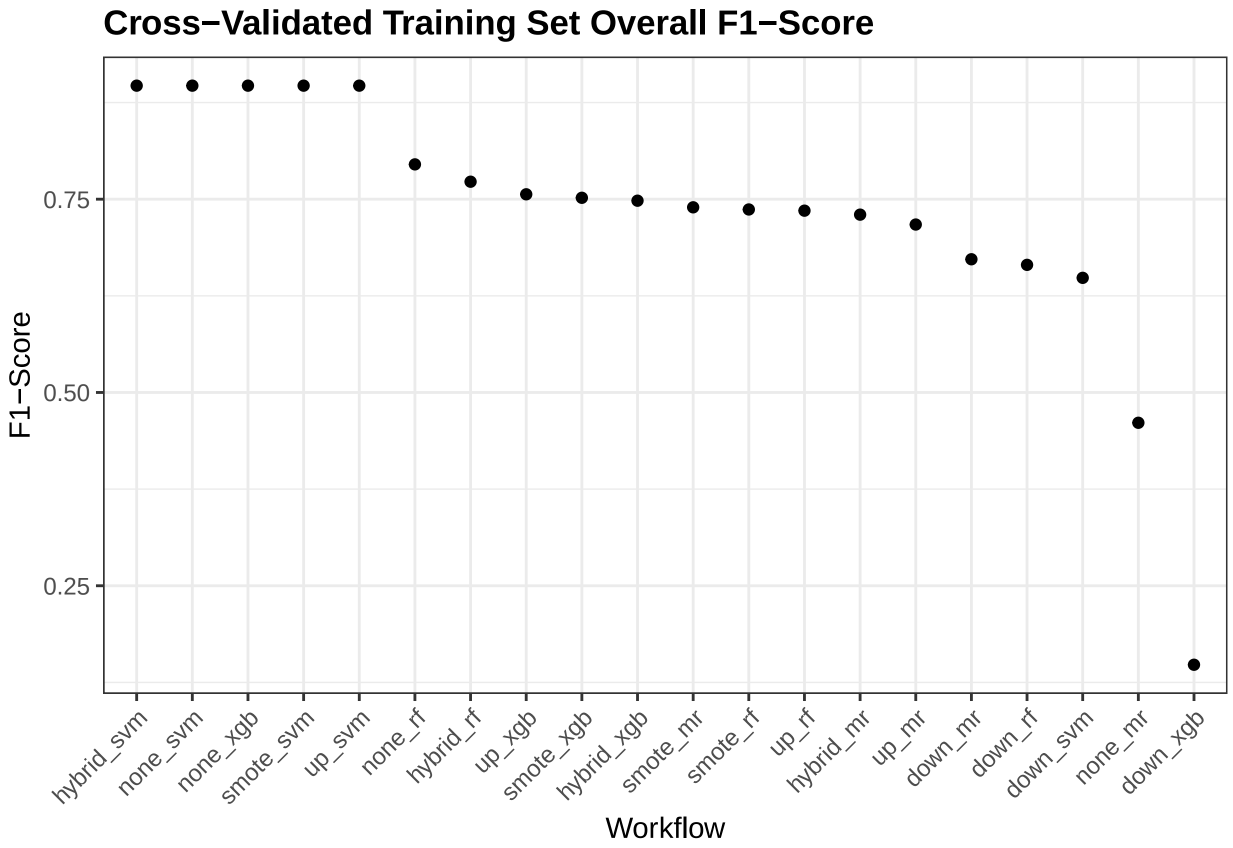


Figure 4.3: Training Set F1-Score

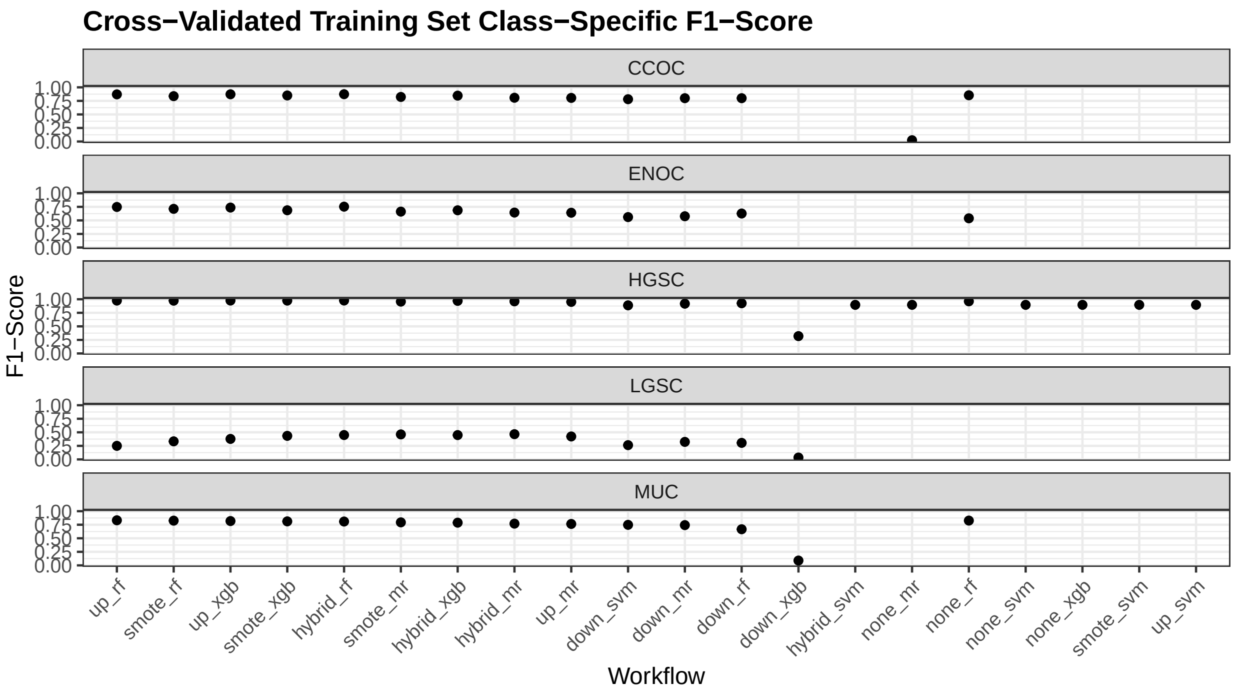


Figure 4.4: Training Set Class-Specific F1-Score

Table 4.4: Cross-Validated Training Set Class-Specific F1-Score

samp	histotype	mr	rf	svm	xgb
none	CCOC	0.024	0.853	NA	NA
none	ENOC	NA	0.538	NA	NA
none	HGSC	0.897	0.962	0.897	0.897
none	LGSC	NA	NA	NA	NA
none	MUC	NA	0.826	NA	NA
down	CCOC	0.8	0.8	0.781	NA
down	ENOC	0.577	0.627	0.561	NA
down	HGSC	0.918	0.926	0.888	0.32
down	LGSC	0.324	0.305	0.263	0.035
down	MUC	0.743	0.667	0.748	0.089
up	CCOC	0.805	0.87	NA	0.872
up	ENOC	0.642	0.748	NA	0.737
up	HGSC	0.951	0.976	0.897	0.976
up	LGSC	0.424	0.25	NA	0.378
up	MUC	0.765	0.832	NA	0.818
smote	CCOC	0.823	0.837	NA	0.85
smote	ENOC	0.662	0.713	NA	0.687
smote	HGSC	0.957	0.974	0.897	0.975
smote	LGSC	0.462	0.333	NA	0.435
smote	MUC	0.794	0.825	NA	0.812
hybrid	CCOC	0.808	0.873	NA	0.847
hybrid	ENOC	0.645	0.754	NA	0.687
hybrid	HGSC	0.962	**0.977**	0.897	0.97
hybrid	LGSC	0.466	0.45	NA	0.449
hybrid	MUC	0.77	0.809	NA	0.788

Table 4.5: Cross-Validated Training Set Overall Kappa

samp	mr	rf	svm	xgb
none	0.007	0.729	0	0
down	0.629	0.638	0.569	-0.002
up	0.717	0.811	0	0.814
smote	0.743	0.797	0	0.799
hybrid	0.744	**0.82**	0	0.783

4.1.3 Kappa

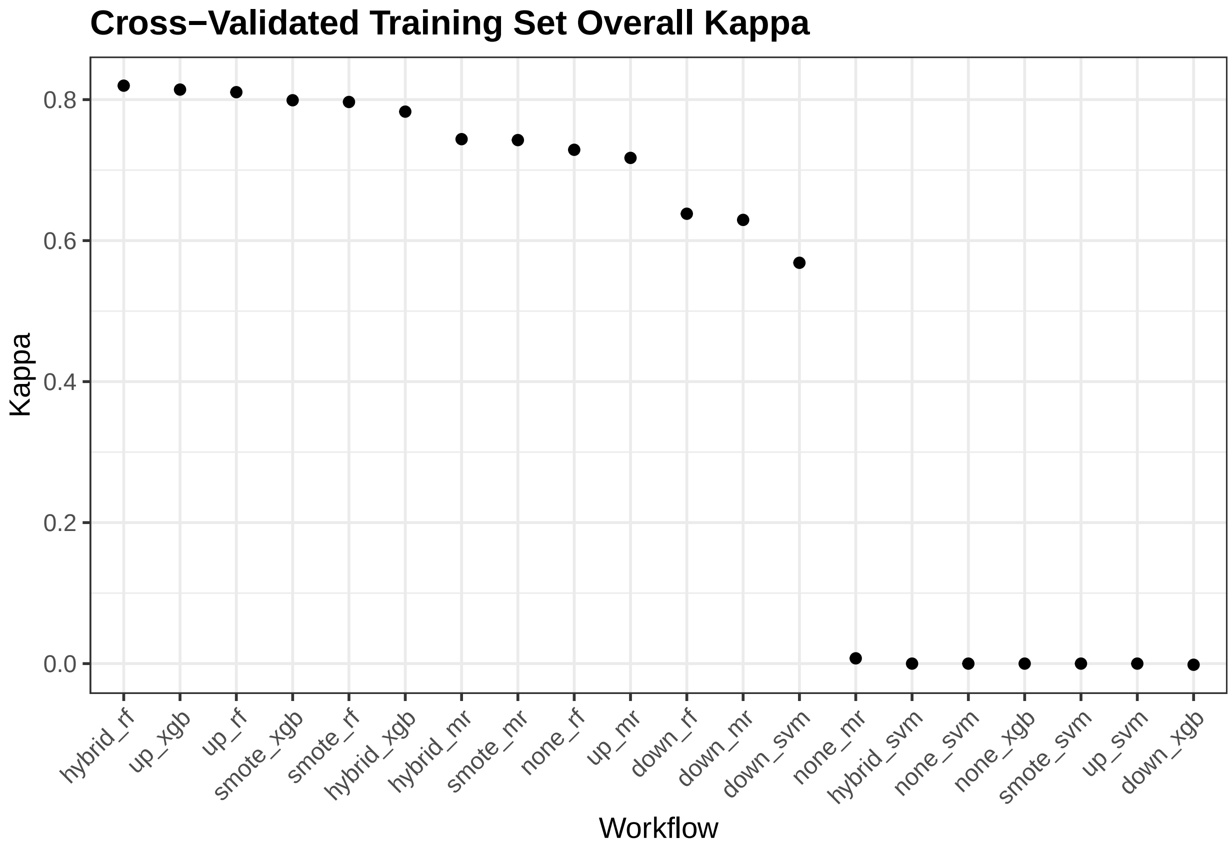


Figure 4.5: Training Set Kappa

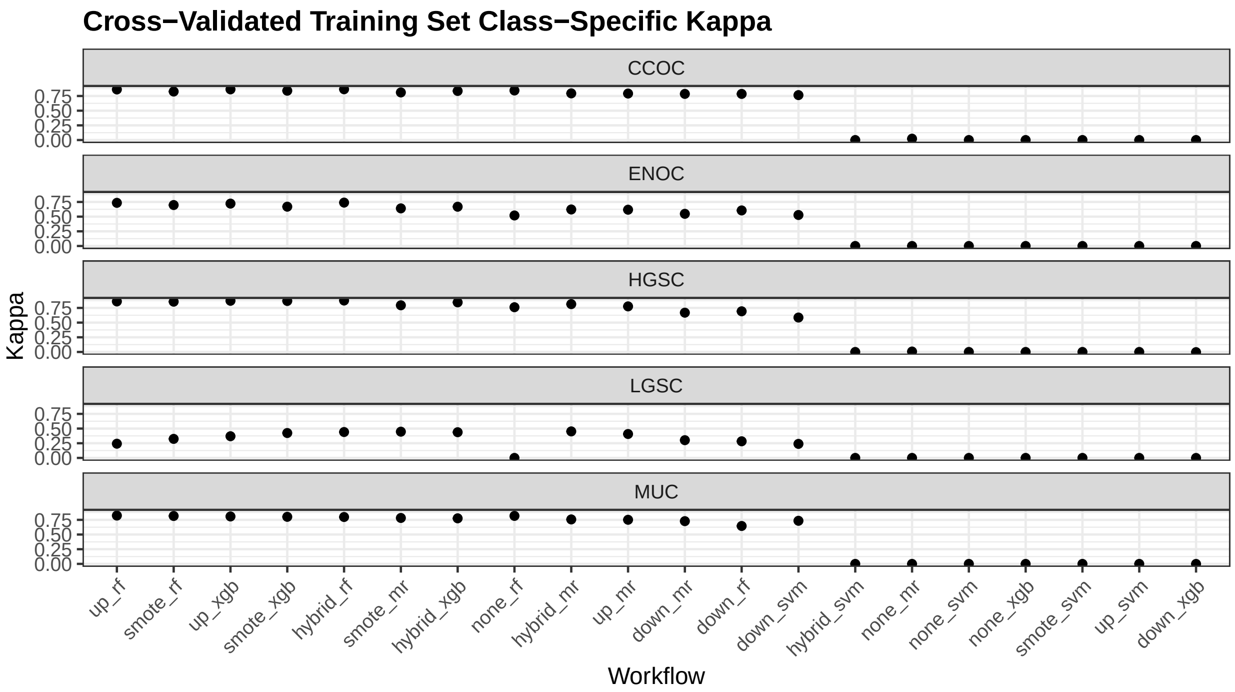


Figure 4.6: Training Set Class-Specific Kappa

Table 4.6: Cross-Validated Training Set Class-Specific Kappa

samp	histotype	mr	rf	svm	xgb
none	CCOC	0.023	0.844	0	0
none	ENOC	0	0.52	0	0
none	HGSC	0.007	0.762	0	0
none	LGSC	0	0	0	0
none	MUC	0	0.818	0	0
down	CCOC	0.786	0.785	0.765	0
down	ENOC	0.548	0.606	0.528	0
down	HGSC	0.669	0.692	0.586	-0.003
down	LGSC	0.302	0.284	0.239	-0.001
down	MUC	0.728	0.646	0.734	-0.001
up	CCOC	0.792	0.862	0	0.863
up	ENOC	0.618	0.735	0	0.722
up	HGSC	0.776	0.861	0	0.872
up	LGSC	0.408	0.242	0	0.37
up	MUC	0.751	0.823	0	0.808
smote	CCOC	0.811	0.826	0	0.84
smote	ENOC	0.641	0.698	0	0.67
smote	HGSC	0.795	0.857	0	0.869
smote	LGSC	0.447	0.324	0	0.424
smote	MUC	0.782	0.816	0	0.801
hybrid	CCOC	0.795	0.865	0	0.836
hybrid	ENOC	0.622	0.739	0	0.669
hybrid	HGSC	0.815	**0.876**	0	0.845
hybrid	LGSC	0.452	0.441	0	0.438
hybrid	MUC	0.757	0.799	0	0.776

Table 4.7: Cross-Validated Training Set Overall G-mean

samp	mr	rf	svm	xgb
none	0.111	0.717	**1**	**1**
down	0.829	0.8	0.829	0.314
up	0.816	0.606	**1**	0.693
smote	0.824	0.656	**1**	0.73
hybrid	0.801	0.736	**1**	0.743

4.1.4 G-mean

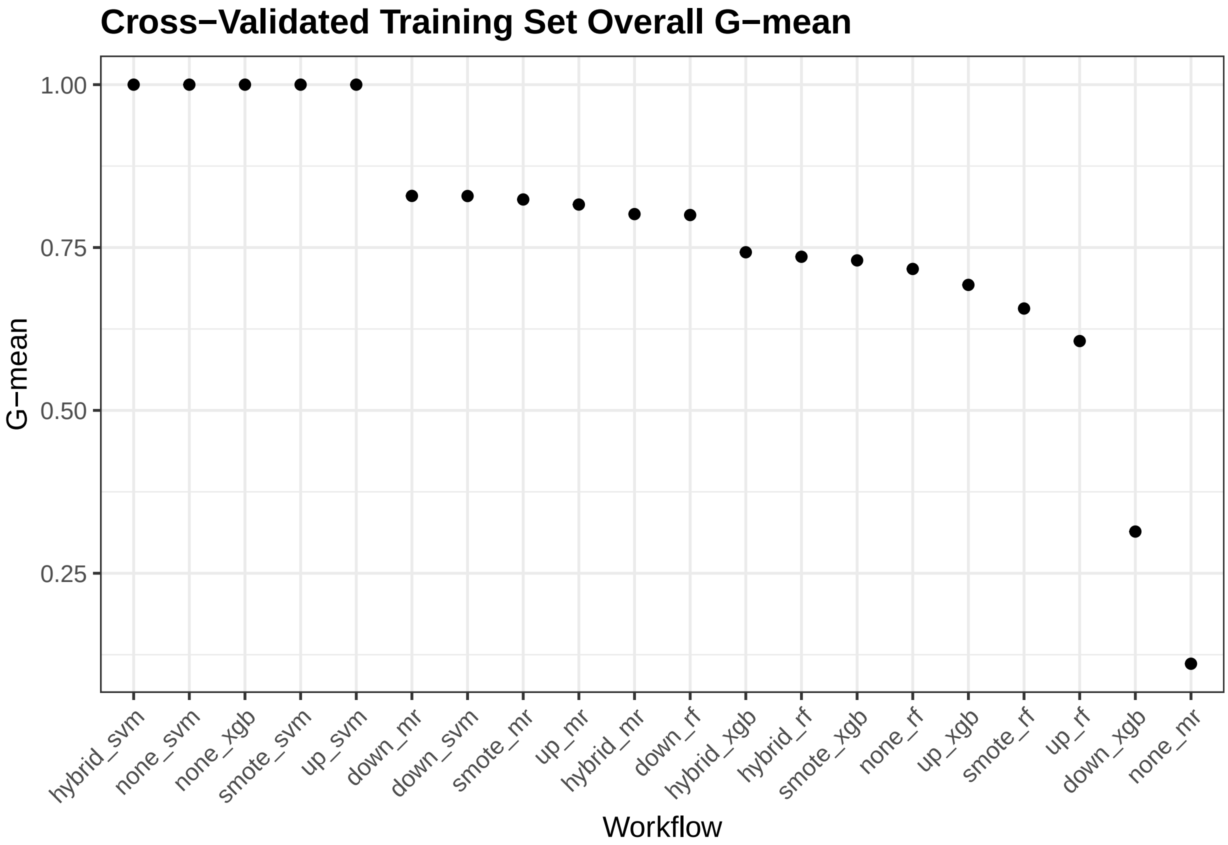


Figure 4.7: Training Set G-mean

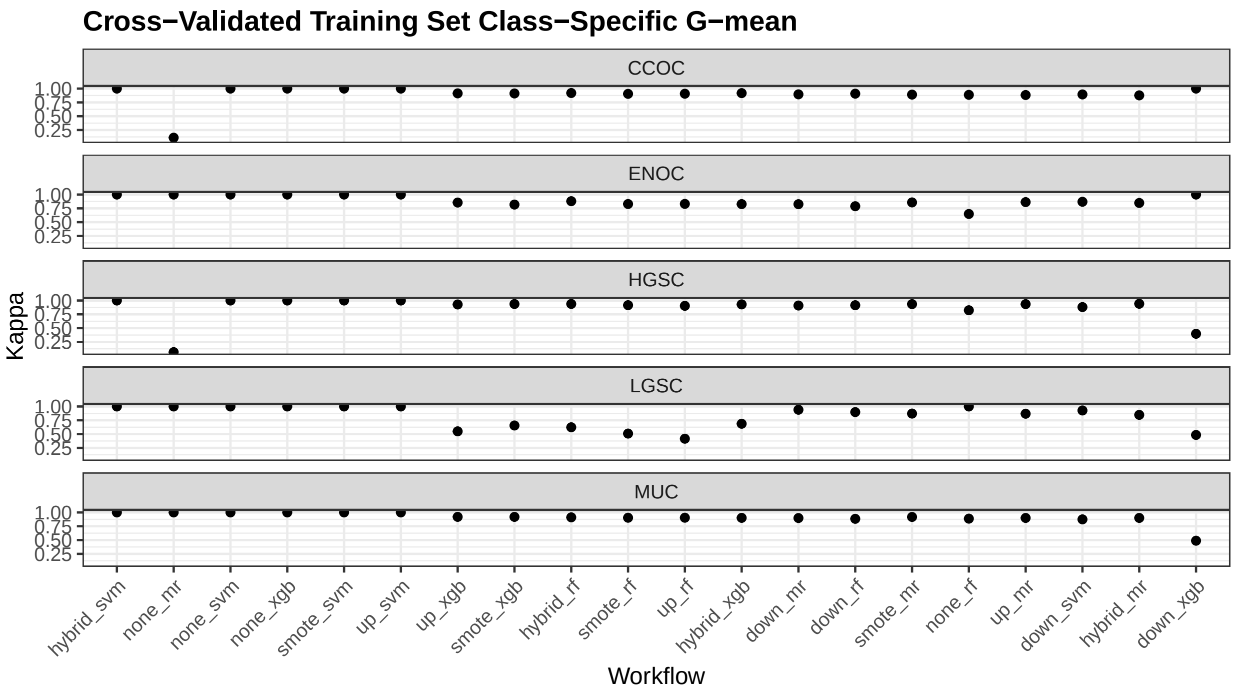


Figure 4.8: Training Set Class-Specific G-mean

4.2 Gene Optimization

4.2.1 Overlap with Other Sets

There are 16 genes out of the 72 common 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 13 genes out of the 72 classifier set that overlap with the SPOT signature: HIF1A, CXCL10, DUSP4, SOX17, MITF, CDKN3, BRCA2, CEACAM5, ANXA4, SERPINE1, TCF7L1, CRABP2, DNAJC9.

4.2.2 Optimal Gene Set

There are 28 unique genes from the combined PrOTYPE and SPOT lists that we want to use for the final classifier. We then incrementally add genes from the remaining 44 candidates based on variable importance scores to this list and recalculate performance metrics. The number of genes at which the performance starts to plateau may indicate an optimal gene set for us to carry forward for a particular model.

Variable importance is calculated using either a model-based approach if it is available, or a SHAP-based VI score otherwise (e.g. for SVM). For the sequential and two-step classifiers, we calculate overall VI scores by aggregating the base classifier VI scores using rank aggregation.

Table 4.8: Cross-Validated Training Set Class-Specific G-mean

samp	histotype	mr	rf	svm	xgb
none	CCOC	0.111	0.887	**1**	**1**
none	ENOC	**1**	0.649	**1**	**1**
none	HGSC	0.066	0.824	**1**	**1**
none	LGSC	**1**	**1**	**1**	**1**
none	MUC	**1**	0.888	**1**	**1**
down	CCOC	0.896	0.908	0.894	**1**
down	ENOC	0.826	0.789	0.869	**1**
down	HGSC	0.909	0.915	0.881	0.398
down	LGSC	0.941	0.898	0.928	0.485
down	MUC	0.899	0.885	0.875	0.488
up	CCOC	0.884	0.907	**1**	0.913
up	ENOC	0.863	0.831	**1**	0.855
up	HGSC	0.935	0.904	**1**	0.929
up	LGSC	0.869	0.416	**1**	0.55
up	MUC	0.9	0.905	**1**	0.92
smote	CCOC	0.891	0.905	**1**	0.912
smote	ENOC	0.857	0.829	**1**	0.819
smote	HGSC	0.935	0.915	**1**	0.937
smote	LGSC	0.871	0.509	**1**	0.656
smote	MUC	0.918	0.904	**1**	0.92
hybrid	CCOC	0.877	0.92	**1**	0.918
hybrid	ENOC	0.848	0.88	**1**	0.827
hybrid	HGSC	0.941	0.938	**1**	0.93
hybrid	LGSC	0.848	0.623	**1**	0.687
hybrid	MUC	0.901	0.911	**1**	0.902

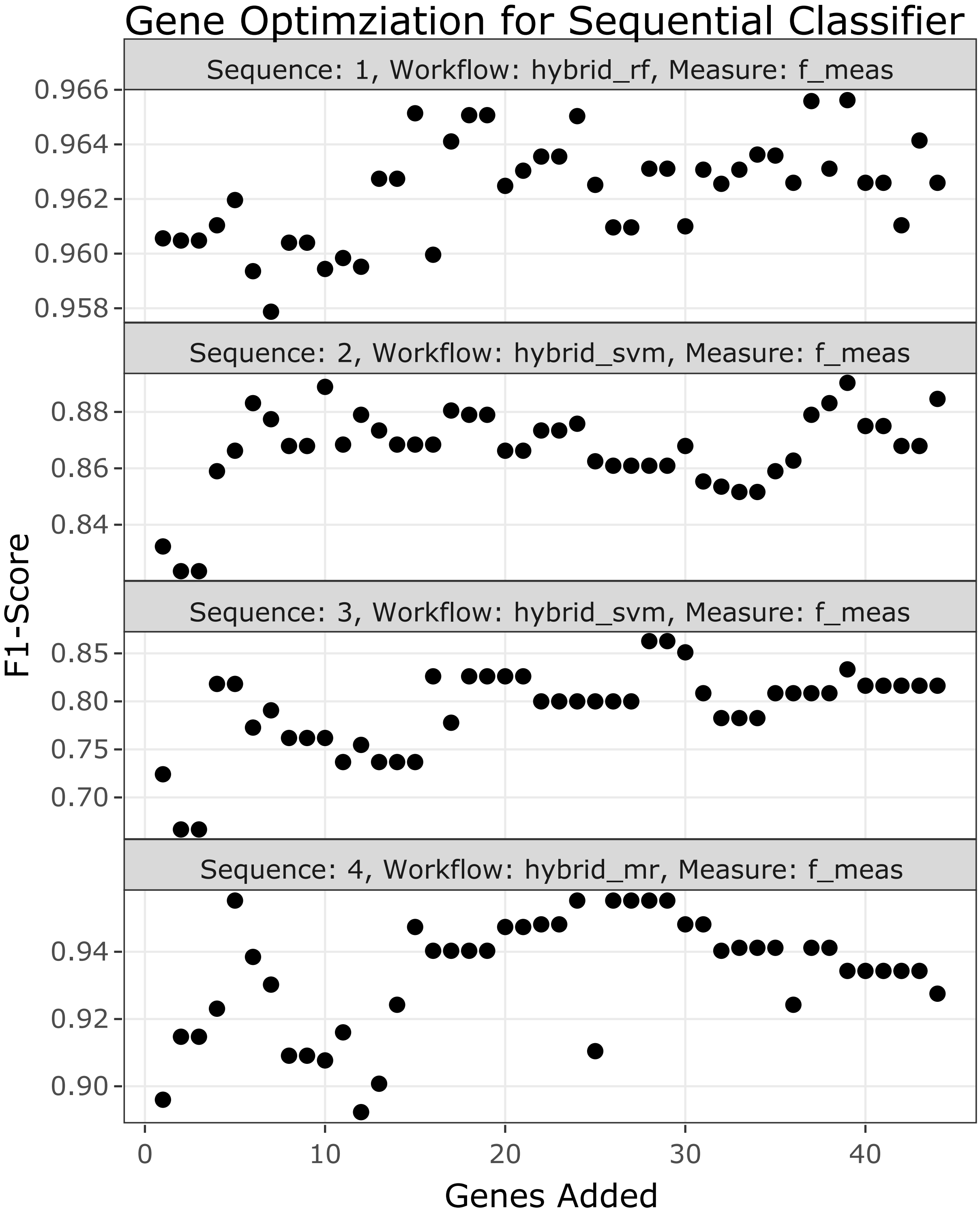


Figure 4.9: Gene Optimization for Sequential Classifier

In the sequential classifier, we use the per-class median F1-scores pertaining to the histotype that had the best performance from each retraining, and sort them on number of genes added. For instance, in sequence 2, we look at the CCOC F1-scores because CCOC had the best performance from retraining after HGSC was removed.

We can observe that in sequence 3, the F1-score stabilizes at around 0.93 when we reach 28 genes added, hence the optimal number of genes used will be n=28+34=62. The added genes are: STC1, TPX2, KGFLP2, MUC5B, CPNE8, HNF1B, BCL2, SLC3A1, ATP5G3, EGFL6, C1orf173, IGFBP1, CYP2C18, FUT3, WT1, KLK7, C10orf116, PBX1, IGJ, DKK4, ZBED1, TP53, LIN28B, GCNT3, MAP1LC3A, MET, GPR64 and SENP8.

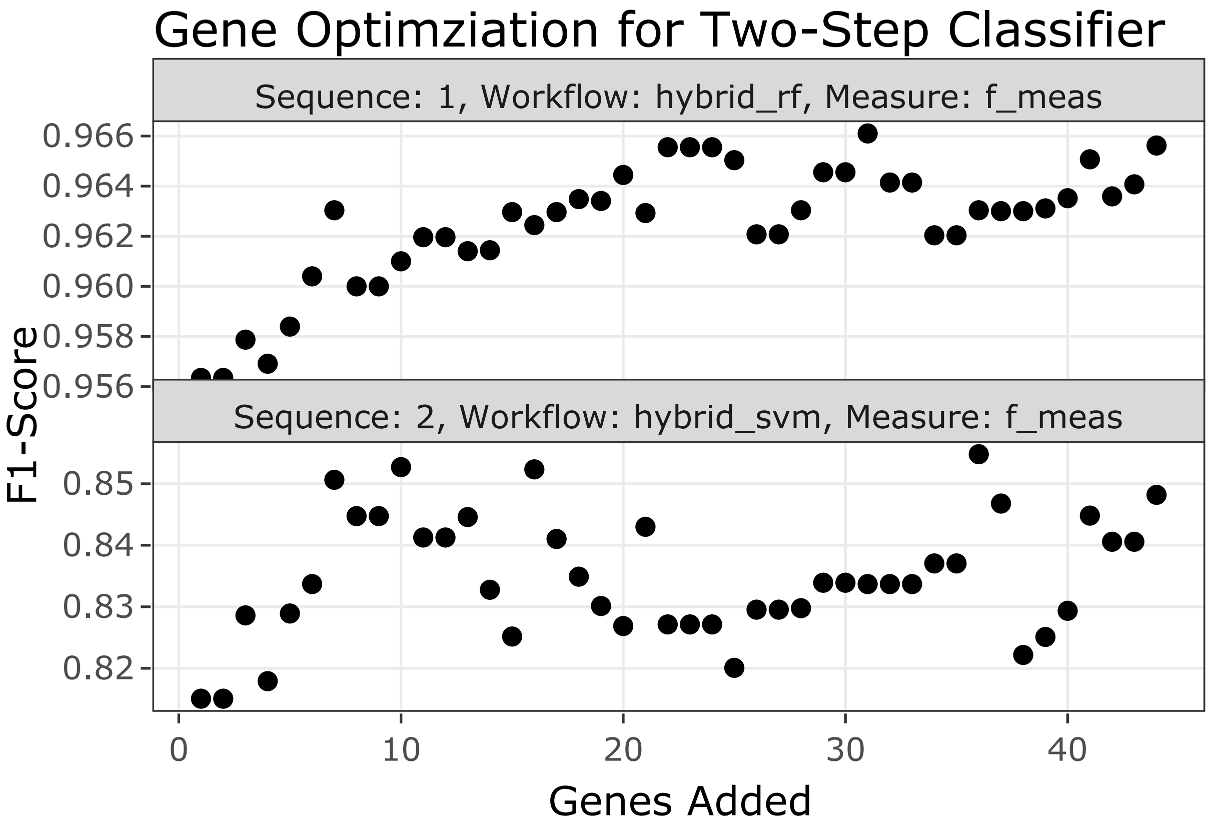
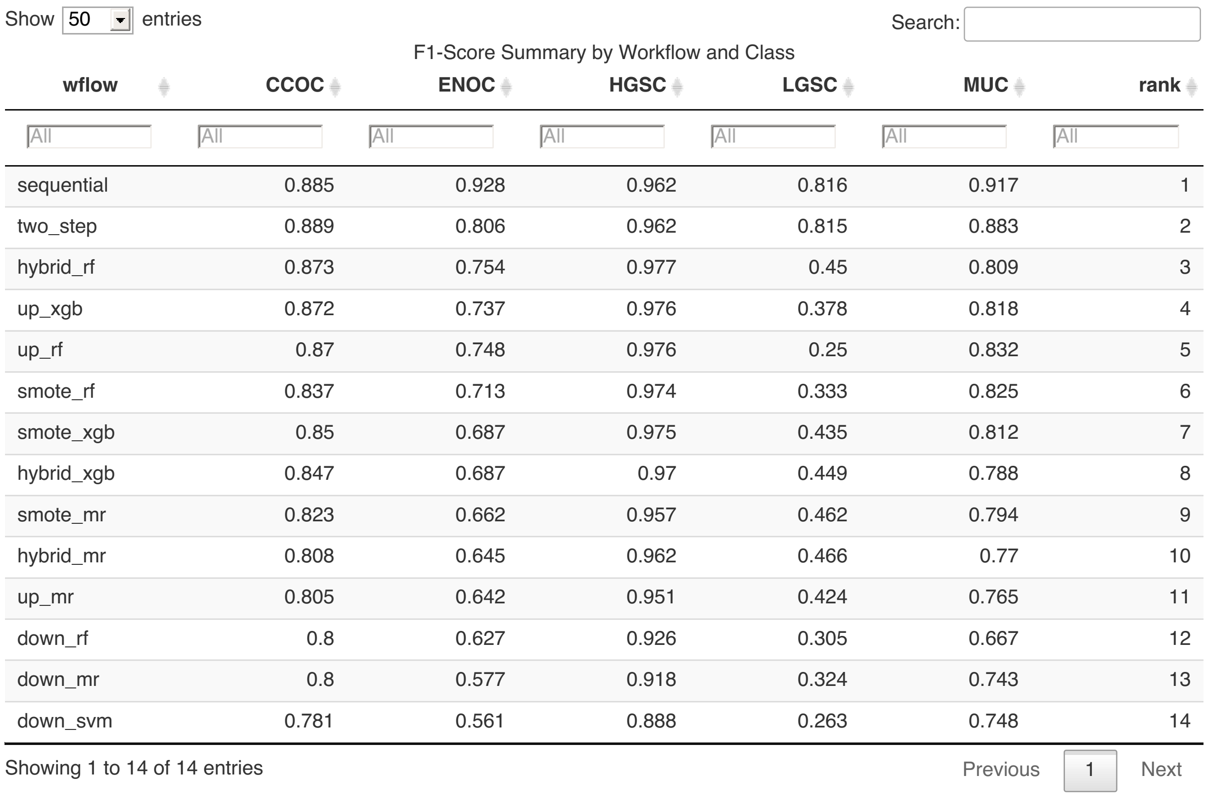


Figure 4.10: Gene Optimization for Two-Step Classifier

Since the second step of the classifier fits a multinomial model, we use the macro F1-score as the measure to analyze gene entry. In the two-step classifier, we see that in Step 2, the F1-score stabilizes at around 0.88 when we reach 31 added. The optimal number of genes used will be n=28+31=52. The added genes are: WT1, KLK7, MUC5B, TFF3, GAD1, TSPAN8, HNF1B, C1orf173, FUT3, STC1, TPX2, TFF1, DKK4, CAPN2, CYP4B1, CPNE8, SLC3A1, KGFLP2, EGFL6, SERPINA5, TP53, CYP2C18, GCNT3, GPR64, ATP5G3, MET, IL6, SEMA6A, LGALS4, ADCYAP1R1 and C10orf116.

4.3 Rank Aggregation



F1-Score Summary by Workflow and Class

wflow	CCOC	ENOC	HGSC	LGSC	MUC	rank
All	All	All	All	All	All	All
sequential	0.885	0.928	0.962	0.816	0.917	1
two_step	0.889	0.806	0.962	0.815	0.883	2
hybrid_rf	0.873	0.754	0.977	0.45	0.809	3
up_xgb	0.872	0.737	0.976	0.378	0.818	4
up_rf	0.87	0.748	0.976	0.25	0.832	5
smote_rf	0.837	0.713	0.974	0.333	0.825	6
smote_xgb	0.85	0.687	0.975	0.435	0.812	7
hybrid_xgb	0.847	0.687	0.97	0.449	0.788	8
smote_mr	0.823	0.662	0.957	0.462	0.794	9
hybrid_mr	0.808	0.645	0.962	0.466	0.77	10
up_mr	0.805	0.642	0.951	0.424	0.765	11
down_rf	0.8	0.627	0.926	0.305	0.667	12
down_mr	0.8	0.577	0.918	0.324	0.743	13
down_svm	0.781	0.561	0.888	0.263	0.748	14

Showing 1 to 14 of 14 entries

The 14 workflows are ordered in the table by their aggregated ranks using the Genetic Algorithm. We see that the best performing methods involve the sequential and two-step algorithms.

4.3.1 Top Workflows

We look at the per-class evaluation metrics of the top 4 workflows.

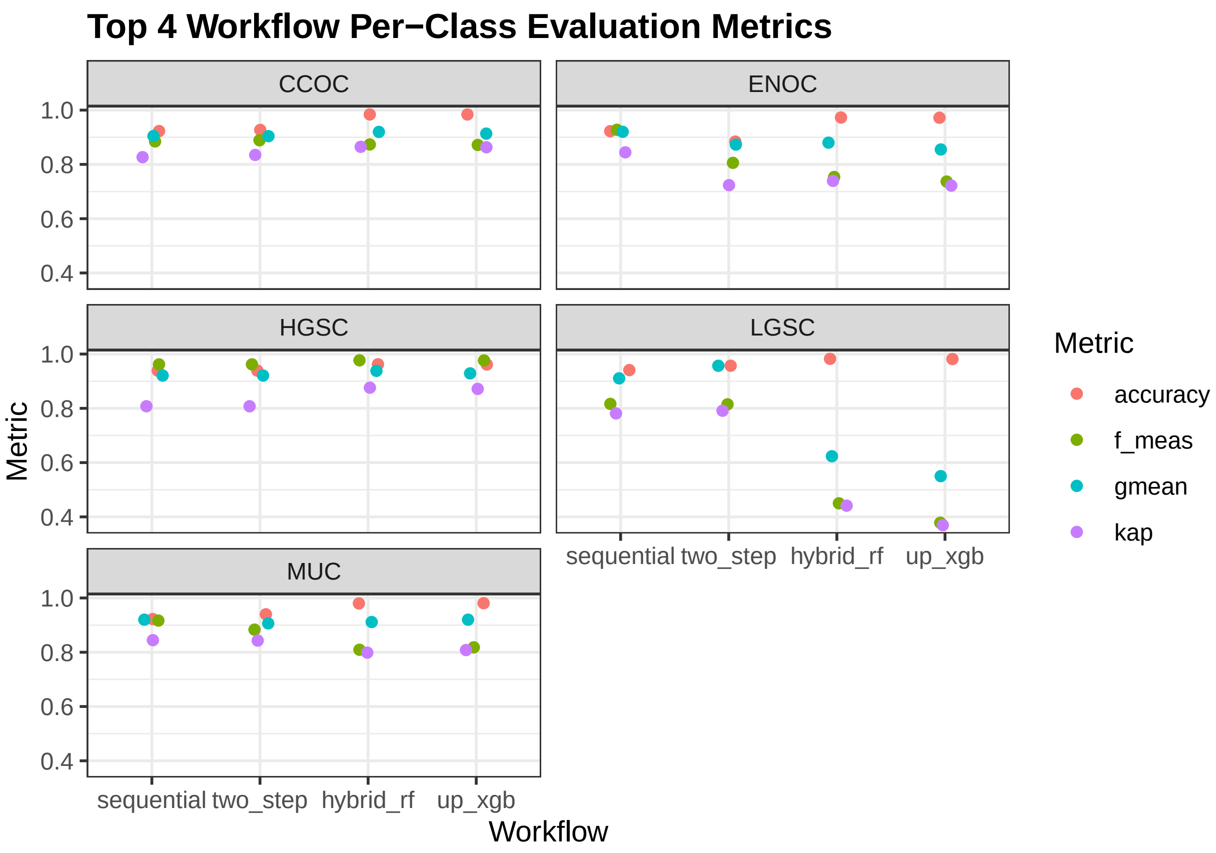


Figure 4.11: Top 4 Workflow Per-Class Evaluation Metrics

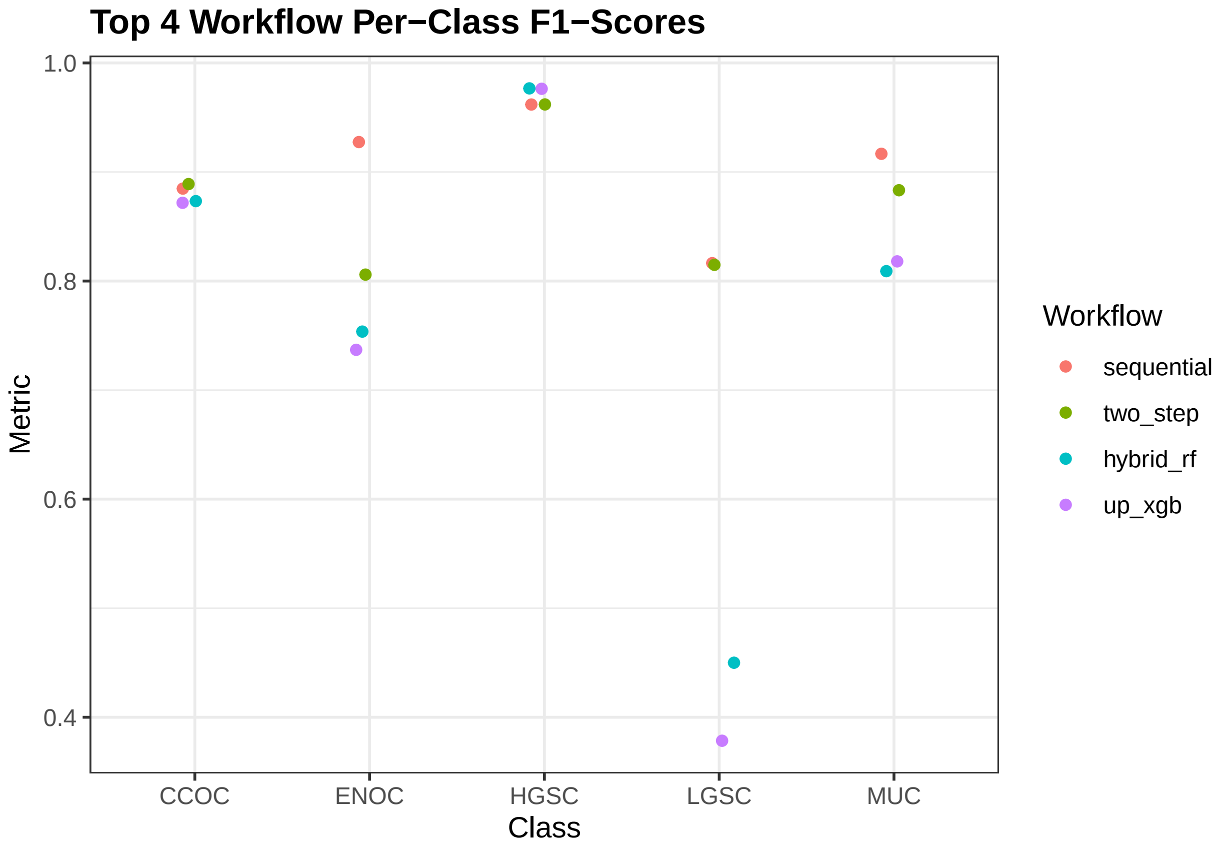


Figure 4.12: Top 4 Workflow Per-Class F1-Scores

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.

4.4 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:

- **sequential**: sequential algorithm with hybrid subsampling at every step. The sequence of algorithms used are:
  - HGSC vs. non-HGSC using random forest
  - CCOC vs. non-CCOC using support vector machine
  - LGSC vs. non-LGSC using support vector machine
  - ENOC vs. MUC using regularized multinomial regression
- **two_step**: two-step algorithm with hybrid subsampling at both steps. The sequence of algorithms used are:
  - HGSC vs. non-HGSC using random forest

Table 4.9: Overall Evaluation Metrics on Confirmation Set Models

method	accuracy	kappa	f1	gmean
sequential_full	0.834	0.669	0.654	0.574
sequential_optimal	0.830	0.666	0.655	0.605
two_step_full	0.840	0.682	0.688	0.650
two_step_optimal	0.844	0.692	0.703	0.657

Table 4.10: Per-Class Eevaluation Metrics on Confirmation Set Model

method	.metric	CCOC	ENOC	HGSC	LGSC	MUC
two_step_full	accuracy	0.970	0.896	0.869	0.969	0.975
	f_meas	0.872	0.626	0.904	0.333	0.704
	kap	0.856	0.568	0.701	0.318	0.691
	gmean	0.924	0.715	0.833	0.614	0.833
two_step_optimal	accuracy	0.963	0.899	0.874	0.972	0.981
	f_meas	0.844	0.645	0.907	0.357	0.760
	kap	0.823	0.588	0.712	0.343	0.750
	gmean	0.919	0.733	0.841	0.615	0.836
sequential_full	accuracy	0.961	0.893	0.869	0.969	0.975
	f_meas	0.839	0.619	0.904	0.231	0.680
	kap	0.817	0.558	0.701	0.215	0.667
	gmean	0.919	0.714	0.833	0.477	0.790
sequential_optimal	accuracy	0.956	0.894	0.871	0.967	0.972
	f_meas	0.821	0.622	0.904	0.276	0.654
	kap	0.796	0.563	0.706	0.259	0.639
	gmean	0.910	0.715	0.839	0.549	0.788

– CCOC vs. ENOC vs. MUC vs. LGSC support vector machine

We can test 2 additional methods by using either the full set of genes or the optimal set of genes for both of these methods.

4.4.1 Confirmation Set

4.4.2 Validation Set

Table 4.11: Overall Evaluation Metrics on Validation Set Model

method	accuracy	kappa	f1	gmean
two_step_optimal	0.875	0.726	0.714	0.776

Table 4.12: Per-Class Eevaluation Metrics on Validation Set Model

method	.metric	CCOC	ENOC	HGSC	LGSC	MUC
two_step_optimal	accuracy	0.975	0.943	0.896	0.976	0.961
	f_meas	0.869	0.752	0.928	0.476	0.545
	kap	0.855	0.720	0.742	0.464	0.527
	gmean	0.963	0.843	0.892	0.739	0.883