Advances in Artificial Intelligence and Machine Learning; Research 4 (1) 1834-1846 Received 12-05-2023; Accepted 23-01-2024; Published 30-01-2024

Efficient Use of Data for Prediction and Validation

Lu Liu
Department of Biostatistics and Bioinformatics,
Duke University Durham, North Carolina
USA

Sin-Ho Jung sinho.jung@duke.edu
Department of Biostatistics and Bioinformatics,
Duke University Durham, North Carolina
USA

Corresponding Author: Sin-Ho Jung

Abstract

Prediction model building is one of the most important tasks in analysis of high-dimensional data. A fitted prediction model should be validated for future use. So, when conducting such an analysis, we have to use the whole data for both training and validation. When using a hold-out method, the fitted prediction model will be more efficient if the training set is bigger, but the validation power will be lower with a smaller validation set. In order to balance the efficiency of fitted prediction model and its validation, 50-50 allocation of the whole data set is popularly used as a hold-out method. In prediction and validation procedure, we have to use the information embedded in the whole data set as efficiently as possible. As a such effort, cross-validation methods (CV) have been very popular these days. In a CV method, a large portion of the data set is used to train models and the remaining small portion of the data is used for validation, and this procedure is repeated until the whole data set is used for validation. In a CV method, each data point is used for both training and validation, so that as the portion of training set is increased, the efficiency of training will be increased, while the validation power will be decreased due to the increased over-fitting, i.e. more frequent use of each data point for training. As another effort of efficient use of the whole data, we propose to use the whole data set for both training and validation, called 1-fold CV method. By using the whole data to fit a prediction model, training efficiency will be highest, but, by reusing the whole data set for validation, its validation power is expected to be very low. The validation power of CV methods will be estimated by permutation methods. Through extensive simulation and real data studies, we conclude that the newly proposed 1-fold CV method uses the available data set very efficiently.

Keywords: C-index; Cox regression, Logistic regression, Machine learning, Permutation, ROC curve.

Citation: Lu Liu and Sin-Ho Jung. Efficient Use of Data for Prediction and Validation. Advances in Artificial Intelligence and Machine Learning. 2024;4(1):106.

1. INTRODUCTION

One of aims of high-throughput project is developing models to predict the outcome based on predictors, called covariates. For example, in a microarray study, our goal is to develop a model to predict the clinical outcome of patients, such as risk of disease, progression, or overall survival, based on the gene expression level. By predicting the risk level, clinicians can select the best treatment for the patient. For a fitted prediction model to be used for future patients, it should be proved that it really predicts the outcome. Hence, development of a prediction model includes a model fitting procedure, called training, and a verification procure of the fitted model, called validation [1].

When a data set containing outcome and predictors from numbers of subjects is given, we have to partition it to form two sets, one for training and the other for validation. A machine learning method is applied to the training set to fit a regression or classification tree model, and we conduct a statistical test to see if the fitted model really predicts the outcomes of the validation set. For a successful prediction model development, we need an efficient model fitting (or training) method to construct a model with high prediction accuracy and a powerful statistical testing method to validate the developed model. Once selecting an efficient machine learning method for training models and a powerful testing method for validation, we have to determine how we partition the given data into training and validation sets for successful prediction model development.

For hold-out methods, the prediction model is more efficient if the training set is bigger, while the validation power will be sacrificed with a smaller validation set. In order to balance the efficiency of fitted prediction model and its validation, 50-50 allocation of the whole data set is popularly used as a hold-out method. However, there is no guarantee that 50-50 hold-out is an optimal partitioning for successful prediction model building.

As an effort for efficient use of the whole data for training and validation, cross-validation (CV) methods [2–4], have been very popularly used these days. In a CV method, a large portion of the whole data set is used to train models and the remaining small portion of the data is used for validation. This procedure is repeated until all data points are used for validation. Therefore, each data point is used for both training and validation. As the portion of training set is increased, the efficiency of training will be increased, but the validation power will be decreased due to the increased over-fitting by using each data point more frequently for training.

One of the issues with these prediction-validation methods is what should be the final prediction model for future use. For a hold-out method, it is not obvious which model should be reported as the final prediction model between the model fitted using the training set and validated using the validation set or the one fitted using the whole data set combining the training and validation sets. One may be hesitant to use the latter since it is not really validated, whereas the former is less efficient since it is developed by using partial data. In a K-fold cross-validation, K prediction models are fitted from K training sets, so that we have a similar issue when selecting a prediction model among K for future use. Furthermore, it is not clear which model is validated by a K-fold cross-validation method.

To overcome this dilemma and as another effort of efficient use of the whole data, we propose to use the whole data set for both training and validation, called 1-fold CV method. By using the whole

data to fit a prediction model, training efficiency will be maximized, but, by reusing the whole data set for validation, its validation power is expected to be low. With the efficient prediction model, though, the lowered validation power due to the small sample size will be somewhat recovered. For the CV methods and 1-fold CV method that use the all data points for both training and validation, the null distribution of the test statistic for validation is derived by permutations.

Multiple literatures compare the performance of various prediction models. Khera, et al. [5], compare the discrimination performance of three machine learning models (XGBoost, neural network and meta-classifier model) with logistic regression using a large cohort study in American College of Cardiology Chest Pain-MI Registry. Christodoulou, et al. [6], conduct a systematic review over 927 studies to investigate the performance of popular machine learning models and logistic regression to predict clinical outcomes. Both of them find no performance benefit of machine learning models over logistic regression.

While many researchers have been intensively assessed the existing prediction methods, there are few studies investigating the performance of validation methods. Diana and Tommasi [7], compare the performance of various cross-validation methods in principal component analysis. Xu and Goodacre [8], compare the performance of various data-splitting methods by the correct classification rate in the test sets. Kim [9], uses classification error rate to evaluate the performance of 10-fold cross-validation and bootstrap methods in the binary classification problem. Borra and Ciaccio [10], conduct simulation studies to compare the performance of various validation methods including K-fold CV, hold-out and bootstrap methods using the prediction error as measurement. Isler, et al. [11], compare the performance of K-fold CV ($K = 2, 3, 5, 10$) and leave-one-out CV on measuring five distinct classifiers in the diagnosis of the patients with congestive heart failure using the heart rate variability data from the normal sinus rhythm and congestive heart failure RR interval databases from the MIT/BIH database in PhysioNET. The previous studies use prediction performance (classification error) as measurements to compare validation methods, but we can not decide that the validation method which achieves the lowest classification error is the best because it may contain overfitting issues. So, in our paper instead of prediction accuracy, we propose a statistical approach to quantify and compare their validation performance of detecting significant relationship between outcome and covariates. Few previous studies talk about determination of the final model, where most of them use an averaged prediction model or an unvalidated model as the final model. Most of the previous studies only focus on classification problems, and in the previous simulation and real data studies, only low-dimensional data sets are considered where the number of covariates is around 10. Also, prediction and validation are combined, so different validation methods have different variable selection power. Previous studies do not care about variable selection of different validation methods. Motivated by them, in our paper, we propose 1-fold CV to solve the dilemma of determining final models, and we consider high-dimensional data sets and compare the variable selection performance of various validation methods. Lastly, we consider both binary and survival data with censoring in our paper.

Pang and Jung [12], propose a permutation method and construct a statistic to decide on model validation using p-value of the log-rank test for survival outcomes. We extend their approach by considering both binary and survival outcomes to investigate which training-validation methods use a given data set most efficiently. We will compare our 1-fold CV method with existing validation methods. Liu, et al. [13], show that the traditional regression models combined with stepwise forward variable selection procedure perform well even with high dimensional data. So, we use their

method for training. For validation, we consider the hold-out, K-fold cross-validation (CV), leave-one-out cross-validation (LOOCV), and our proposed 1-fold CV methods. Since 1-fold CV method and CV methods use all data points for both training and validation, they are incorporated with permutations to remove the over-fitting [14], for the significance testing in validation. We conduct extensive simulation studies to compare the performance of the prediction-validation methods and illustrate them using real data.

2. METHODS

We use the standard regression models with stepwise selection to develop prediction model [13]. We give a brief review of the methods used in our paper and describe how to measure the performance of training and validation.

For totally n subjects, we observe y as outcome variable and $(x_1, ..., x_m)$ as m covariates from each sample. Then the data set looks like $\{y_i, (x_{1i}, ..., x_{mi}), i = 1, ..., n\}$. For high-dimensional data, the size of m is much larger than n, but most of the time there are only a small number of features which are truly associated with the outcome and we denote the number to be m_1.

2.1 Prediction

Let $Z = (z_{\tilde{1}}, ..., z_{\tilde{k}})^T$ denote features that are possibly associated with outcome y, and $\beta = (\beta_{\tilde{1}}, ..., \beta_{\tilde{k}})^T$ their regression coefficients.

2.1.1 Logistic regression

When we have a binary outcome and want to associate it with features, logistic regression is popularly used [15]. For the ith subject $(i = 1, ..., n)$ with covariate Z_i, $p_i(\beta_0, \beta) = P(y_i = 1|Z_i)$ denote the probability that the outcome y_i to be 1. A logistic regression model is given as

$$\log \frac{p_i(\beta_0, \beta)}{1 - p_i(\beta_0, \beta)} = \beta_0 + \beta^T Z_i$$

where β_0 is the intercept. Since the outcome y_i follows independent Bernoulli distribution with success probability p_i when covariates Z_i are given, the log-likelihood is formulated as

$$\ell_1(\beta_0, \beta) = \sum_{i=1}^{n} [y_i \log p_i(\beta_0, \beta) + (1 - y_i) \log\{1 - p_i(\beta_0, \beta)\}]$$

and the coefficients $(\hat{\beta}_0, \hat{\beta}_{\tilde{1}}, ..., \hat{\beta}_{\tilde{k}})$ are estimated by maximizing $\ell_1(\beta_0, \beta)$.

2.1.2 Cox proportional hazards model

We apply Cox proportional hazards model to modeling survival outcomes. For the ith sample, the minimum of survival and censoring time is denoted as y_i and the event indicator is denoted as

δ_i, where $\delta_i = 1$ for event sample and $\delta_i = 0$ for censoring sample. Then the data set is shown as $\{(y_i, \delta_i), (z_{1i}, ..., z_{mi}), i = 1, ..., n\}$. We assume that the censoring time is independent of the survival time given the covariates for each subject, and by the proportional hazard assumption the hazard function $h_i(t)$ is given as

$$h_i(t) = h_0(t)e^{\beta^T Z_i}$$

for subject i at time $t(\geq 0)$, where $h_0(t)$ is the baseline hazard function. By Cox [16], the partial log-likelihood function is given as

$$\ell_2(\beta) = \sum_{i=1}^{n} \delta_i \Big\{ Z_i - \frac{\sum_{j=1}^{n} I(y_j \geq y_i) Z_j \exp(\beta^T Z_j)}{\sum_{j=1}^{n} I(y_j \geq y_i) \exp(\beta^T Z_j)} \Big\}$$

where $I(\cdot)$ is the indicator function and the regression coefficients are estimated by maximizing $\ell_2(\beta)$.

2.1.3 Stepwise forward variable selection

Variable selection (or dimension reduction) is one of the challenges in regression model fitting with high-dimensional data. Forward procedure, backward elimination procedure and all possible combination procedure are popularly applied for variable selection. However, backward elimination and all possible combination procedures are not feasible for high dimensional data because the model estimation does not converge with a large number of covariates. Liu, et al. [13], show that standard regression methods combined with stepwise forward variable selection work well even with high dimensional data, such as microarray data. The selection procedure starts with an empty model (or intercept only), and then the covariate with the smallest p-value at each step is added to the model if it is smaller than a prespecified value α_1, and the covariates are removed if any of them turns insignificant after adding a new feature (p-value larger than another prespecified value α_2). The selection proceeds until no more covariates can be added to the model. We control the number of selections by specifying various values of α_1 for insertion and α_2 for deletion and most of the time α_1 is smaller than α_2.

Existing stepwise programs in R and SAS use penalized likelihood criteria such as Akaike information criterion (AIC) and Bayesian information criterion (BIC) instead of conducting hypothesis testing. One of the shortcomings is that we do not know the significance of the selected features and can not control the number of selections by using these programs.

We denote logistic regression combined with stepwise variable selection as L-SVS and Cox's regression combined with stepwise variable selection as C-SVS.

2.2 Validation

We use L-SVS and C-SVS as the prediction methods to select covariates and fit prediction models, and combine them with various validation methods to assess the accuracy of the fitted models. The validation methods we consider are described below.

2.2.1 Hold-out method

A hold-out method randomly splits the whole data set into a training set and a validation set with a certain allocation proportion. We consider 50-50 and 70-30 allocations in our numerical studies. Prediction model is fitted using the training set and the fitted model is validated using the validation set. Since the held out validation set is never used for training, there is no over-fitting issue, so that we do not need permutations to remove over-fitting in validation.

2.2.2 K-fold CV

The K-fold CV is an averaged approach over multiple iterations. It randomly partitions the whole data set into K pieces with equal size. At each iteration, $K-1$ pieces combined are used to train a prediction model and the remaining one piece is used to validate the fitted model. The overall accuracy of the prediction model is the average of the model accuracy in all iterations. In our numerical studies, we consider 5-fold and 10-fold CV that are shown to be highly efficient by Pang and Jung [12]. Note that in K-fold CV, there will be K training steps, each using $K-1$ pieces of data sets, and each data point is included in $K-1$ training steps and one validation. As such, the fitted prediction models are subject to over-fitting which will be taken care by permutations as described in Section 2.3.3.

2.2.3 LOOCV

LOOCV is a special case of K-fold CV where K equals n, the sample size of the whole data set. At each iteration, the whole data set except one sample point is used for training and the one that is held out is used for testing. LOOCV involves n training steps, so that it requires heavy computations, especially with a large sample size. As such, it is more frequently applied to data set with a small sample size. LOOCV also requires permutations for unbiased validation.

2.2.4 One-fold CV

We propose a method, called 1-fold method, that uses the whole data set for both prediction and validation. By 1-fold CV method, a prediction model is fitted using the whole data set, and a test statistic based on an accuracy of the fitted model to predict the outcome of the whole data set. The prediction model will be highly efficient since it is fitted from the whole data set, but the information of the data used for validation based on the accuracy parameter will be small due to the high over-fitting issue by using the same data set twice. We use permutations to generate the null distribution of a validation test statistic based on the accuracy parameter. If the fitted model is validated by the permutation test, the prediction model that is fitted from the whole data set will be reported for future use.

2.3 Performance Measurements

The performance of a fitted prediction model is measured by an accuracy parameter that is estimated from validation set. Different measurements are used for different types of outcome variable. The area under the curve (AUC) of receiver operating characteristic (ROC) is used for binary outcomes and the log-rank test is conducted for survival outcomes as described below.

2.3.1 AUC test for validation of binary outcomes

Let $\hat{\beta}$ denote the vector of regression estimates of the logistic model fitted from the training set. For the validation set (y_i, Z_i), we calculate the risk score $r_i = \hat{\beta}^T Z_i$ of subject i and see how well r_i predicts binary outcome y_i. AUC of ROC curve is popularly used to measure the association between a continuous or discrete variable r_i and a binary outcome y_i. We partition the validation set into a control group with $y_i = 0$, $\{r_{01}, ..., r_{0n_0}\}$, and a case group with $y_i = 1$, $r_{11}, ..., r_{1n_1}\}$. The AUC of the ROC curve for (y_i, Z_i) is $\theta = P(r_{1i} > r_{0j})$ that is estimated by

$$\hat{\theta} = \frac{\sum_{i=1}^{n_0} \sum_{j=1}^{n_1} S(r_{0i}, r_{1j})}{n_0 n_1}$$

where $S(r_{0i}, r_{1j}) = 1$ if $r_{0i} > r_{1j}$, $= 1/2$ if $r_{0i} = r_{1j}$, and $= 0$ if $r_{0i} < r_{1j}$.

If $\theta = 1/2$, then r_i does not predict the outcome y_i at all, and θ close to 1 means subjects with large r_i values have a high chance to have $y_i = 1$. In order to verify the predicting power of the prediction model fitted from the training set, we perform a statistical test for $H_0 : \theta = 1/2$ based on the estimator $\hat{\theta}$ from the validation set. Hanley and McNeil [17], and Mason and Graham [18], show that $\hat{\theta}$ is equivalent to the nonparametric Mann–Whitney [19], U test statistic, so that we can use the critical value for the Mann–Whitney U test when the training and validation sets are independent as in a hold-out validation case. However, in CV and 1-fold CV methods, each data point is used for training and validation, so that we can not use Mann–Whitney critical values due to the over-fitting. In this case, we conduct permutations to derive the null distribution of the U statistic as described in Section 2.3.3. We validate the fitted prediction model if the null hypothesis is rejected for a pre-specified type I error rate.

2.3.2 Log-rank test for validation of survival outcomes

When the outcome is a survival endpoint, a Cox's proportional hazards model is fitted from the training set to predict the outcome. Let $\hat{\beta}$ denote regression estimates of the fitted prediction model and Z denote the vector of corresponding covariates. Using the median of risk scores $r_i = \hat{\beta}^T Z_i$ in the training set, we partition the validation set into a high risk group with risk scores larger than the median and a low risk group with risk scores smaller than the median. The two-sample log-rank test [20], is applied to the validation set to compare the survival distributions between the high and low risk groups.

If the training and validation sets are independent as in a hold-out method, we can use the standard log-rank test by Peto and Peto [20], to validate the fitted prediction model. As in CV and 1-fold

CV cases, however, if each data point is used for both training and validation, then we conduct permutations to derive the null distribution of the log-rank statistic as described in Section 2.3.3.

2.3.3 Permutations for CV methods

Cross-validation methods (including K-fold CV, LOOCV and 1-fold CV) use each data point for both training and validation, so that we conduct permutations to generate the null distribution of test statistic for validation while removing the over-fitting bias.

Let w_0 denote the test statistic, such as the U statistic (or $\hat{\theta}$) for a binary outcome and the log-rank test for a survival outcome calculated from the original data set. At each permutation, we shuffle the outcome variable and randomly match with the covariate vectors. Since the connection between outcome and the covariates, the permuted data satisfied the null hypothesis. We apply the same training and validation methods to the permuted data and calculate the test statistic. Let w_b denote the test statistic calculated from the b-th permutation data. The 2-sided p-value for the validation test is estimated by p-value $= B^{-1} \sum_{b=1}^{B} I(|w_b| \geq |w_0|)$, where B is the total number of validations. We reject the null hypothesis (or validate the fitted prediction model) if p-value is smaller than a pre-specified α value.

3. RESULTS

3.1 Simulation Studies

The main focus of our paper is to compare the performance of different validation methods, while using the prediction method of L-SVS for binary outcomes and C-SVS for survival outcomes. Since training and validation steps are always related, validation methods can affect the prediction performance, so that we also investigate the performance of variable selection when the prediction models are incorporated with different validation methods.

$n = 200$ samples with $m = 1000$ candidate predictors are generated block-wise, which consists 100 independent blocks of block size 10 and within each block the values follow a multivariate Gaussian distribution with means 0 and variances 1. Also, within each block, the multivariate Gaussian distribution has a compound symmetry correlation structure and the correlation coefficient $\rho = 0.1$ for all blocks. We assume that over the $m = 1000$ candidate predictors, only $m_1 = 6$ of them are truly associated with the outcome.

For binary outcome, the outcome y_i of the ith subject ($i = 1, ..., n$) with true predictors $\tilde{z}_i = (z_{\tilde{1}i}, ..., z_{\tilde{m}_1 i})^T$ is generated from Bernoulli distributions with success probability p_i which is calculated by the logistic regression model

$$p_i = P(y_i = 1) = \frac{\exp(\beta_0 + \beta^T \tilde{z}_i)}{1 + \exp(\beta_0 + \beta^T \tilde{z}_i)},$$

where $\beta = (\beta_{\tilde{1}}, ..., \beta_{\tilde{m}_1})^T$ denotes the vector of regression coefficients of the true predictors. The six true predictors are from two different blocks, three of them are from the first block and the other three

are from the second block. The value of regression coefficients are $\beta_{\tilde{l}} = (-1)^{l+1} * 0.7, l = 0, 1, ..., m_1$ and $\beta_{\tilde{0}}$ is the intercept term.

We fit prediction models by applying L-SVS to each training set with $\alpha_1 = 0.002$ for inclusion and $\alpha_2 = 0.004$ for deletion, and validate the fitted prediction model using 50-50 hold-out, 70-30 hold-out, 5-fold CV, 10-fold CV and 1-fold CV methods. We count the true selection (number of true covariates included in the fitted model) and the false selection by the fitted models and calculate the empirical rejection rate of the $H_0 : \theta = 1/2$ by the validation test using U statistic with 2-sided $\alpha = 0.05$. For K-fold CV method, K prediction models are fitted from each of simulated data set, so that true and false selections are averaged over K fitted models.

Both the prediction performance (mean true and false selections) and validation performance (mean p-value of AUC test and empirical power) are estimated and summarized in TABLE 1. Obviously, the efficiency of prediction will be increasing in the size of training set. As a result, among the validation methods, true selection increases as the size of training set gets larger. With a larger training set, false selection also increases, but the increment in false selection is much smaller than that in true selection. By using the whole data set for training, 1-fold CV has the highest training efficiency.

Table 1: Simulation results on a binary outcome

	50-50 hold-out	70-30 hold-out	1-fold CV	5-fold CV	10-fold CV
True Selections	0.9	1.98	3.5	2.448	2.958
False Selections	1.21	1.58	1.98	1.68	1.775
P-value	0.305	0.191	0.038	0.056	0.048
Empirical Power	0.42	0.53	0.77	0.74	0.78

Roughly speaking, the amount of information in validation set is the proportion of independent validation set with respect to the whole data set for hold-out methods. Note that the proportion of independent validation set relative to the whole data set is 0.5, 0.3, $1/5$, $1/10$, and 0 for 50-50 hold-out, 70-30 hold-out, 5-fold CV, 10-fold CV, and 1-fold CV methods, respectively. But, CV methods will have some additional information by recycling the data points for prediction and validation. From TABLE 1, in spite of its lowest information for validation set, 1-fold CV has the smallest mean p-value because of its efficient prediction. In terms of empirical power, however, 1-fold CV and 10-fold CV have similarly high validation power. Combined with L-SVS, 10-fold CV has better prediction-validation performance than 5-fold CV. Hold-out methods have very low prediction and validation performance by not using the data efficiently.

For survival outcomes, the covariate vectors are generated in the same way as in the binary outcome case. For subject $i = 1, ..., n$ with true predictors $z_i = (z_{\tilde{1}i}, ..., z_{\tilde{m}_1 i})^T$, at time t the hazard rate is

$$h_i(t) = h_0 e^{\beta^T z_i}$$

where $\beta = (\beta_{\tilde{1}}, ..., \beta_{\tilde{m}_1})^T$ is the regression coefficients of true predictors z_i. The six true predictors are specified in the same way before and $\beta_{\tilde{l}} = (-1)^{l+1} * 0.4$ for $l = 1, 2, ..., m_1 (= 6)$ and $h_0 = 0.1$. We consider 10% or 30% censoring. For 30% of censoring, censoring times are generated from a uniform distribution $U(0, a)$. And after fixing the accrual period a, an additional follow-up period b is selected to generate 10% of censoring from uniform distribution $U(a, a + b)$.

We apply C-SVS with $\alpha_1 = 0.001$ for inclusion and $\alpha_2 = 0.002$ for deletion to training prediction models, and validate the fitted models using various validation methods. Using the median risk score $r_i = \hat{\beta}^T z_i$ of training set as the cutoff, we partition the validation into high and low risk groups and perform the log-rank test to compare the survival distributions between the two groups. For each censoring proportion, $N = 100$ simulation data sets of size $n = 200$ are generated. The p-value of the log-rank test is estimated by permuting each data set $B = 200$ times for the CV methods.

We calculate the mean true and false selections of fitted prediction models, and mean p-value and empirical power of the validation test. The simulation results are shown in TABLE 2. With a lower (10%) censoring proportion, each validation method has higher prediction performance in terms of true selection and higher validation performance in terms of both mean p-value and power. Note that false selection slightly increases with a lower censoring too. For both 10% or 30% censoring cases, 1-fold CV has the highest prediction performance in terms of true selection, while the hold-out methods have the lowest true selection among the five validation methods. Furthermore, 1-fold CV has the lowest false selection among the five validation methods in 10% censoring case and next to 70-30 hold-out in 30% censoring case. The difference in false selection is smaller than that in true selection among the five methods. In terms of mean p-value and empirical power, 1-fold CV has the highest validation performance for both 10% or 30% censoring cases. By using data inefficiently, hold-out methods have low prediction and validation performance. Combined with C-SVS, 10-fold CV seems to have slightly better prediction-validation performance than 5-fold CV.

Table 2: Simulation results on a survival outcome

	50-50 hold-out	70-30 hold-out	1-fold CV	5-fold CV	10-fold CV
	(i) 30% Censoring				
True Selections	0.83	2.08	3.91	2.646	3.336
False Selections	1.65	1.35	1.42	1.58	1.59
P-value	0.321	0.214	0.035	0.122	0.125
Empirical Power	0.28	0.43	0.84	0.61	0.63
	(ii) 10% Censoring				
True Selections	1.38	2.8	5.22	3.742	4.486
False Selections	1.56	1.91	1.53	1.692	1.655
P-value	0.27	0.16	0.009	0.063	0.034
Empirical Power	0.39	0.58	0.97	0.83	0.82

Since hold-out methods fit only one prediction model and do not require permutations for validation test, their computation time is much shorter than CV methods. Among the three CV methods, the computing of 1-fold validation is much faster than that of 5-fold and 10-fold CV methods. We did not include LOOCV in this simulation study due to its overly heavy computation.

3.2 Real Data Examples

In Farrow NE, et al. [21], the sentinel lymph node (SLN) specimens of 60 patients from a retrospective melanoma cohort are studied and the expression level of 730 immune-related genes in the specimens form the Nanostring nCounter PanCancer Immune Profiling Panel. A considerable proportion of patients experience recurrence of melanoma after surgery, and our goal is to identify

patients with high risk of recurrence and treat them with adjuvant therapy early to eliminate residual disease and improve patient prognosis. In this study, we predict the recurrence-free survival (RFS) using the gene expression and patient characteristics.

We apply C-SVS with $(\alpha_1, \alpha_2) = (0.05, 0.1)$ combined with 50-50 hold-out, 70-30 hold-out, 1-fold CV, 5-fold CV, 10-fold CV and LOOCV to the microarray data with RFS as an outcome. TABLE 3 shows the analysis results. Note that 1-fold CV selects the fewest features (same as 50-50 hold-out), and all of the methods select add_trt as it is significant to predict RFS. Hold-out methods have large p-values for validation, and LOOCV has a smaller p-value than 5-fold and 10-fold CV methods. One-fold CV has the smallest p-value for validation.

Table 3: Analysis results of Farrow, et al. [21], data

Method	# Selected Features	P-value
50-50 hold-out	3	0.426
70-30 hold-out	5	0.616
1-fold CV	3	0.08
5-fold CV	9	0.5
10-fold CV	8.9	0.28
LOOCV	4.673	0.135
	Selected Features (included in any K fitted models for K-fold CV	
50-50 hold-out	add_trt, sex, GE_TPTE	
70-30 hold-out	add_trt, GE_KIR3DL3, GE_HLA_DQA1, GE_SERPINB2, GE_IL19	
1-fold validation	add_trt, GE_NEFL, GE_TMEFF2	
5-fold CV	add_trt, GE_CCR3, GE_CX3CR1, GE_IL5	
10-fold CV	add_trt, GE_TMEFF2, GE_NEFL, GE_IL3	
LOOCV	GE_TMEFF2, add_trt, GE_NEFL, GE_KIR_Activating_Subgroup_1	

All analyses are conducted using open-source R software, version 3.6.0 (R Foundation for Statistical Computing).

4. DISCUSSIONS

Development of a prediction model involves training and validation. Successful development of a prediction model requires efficient use of data for training and validation as well as an efficient model fitting method for training and a powerful test for validation. In this paper, we have investigated how we can efficiently use a given data set for training and validation, while an efficient model fitting method and a powerful validation test are selected. As an effort for efficient use of data, we propose 1-fold CV method. We have compared its performance of training and validation with other popular validations methods for binary and survival outcomes. We conclude that our proposed 1-fold CV method performs better in both training and validation than hold-out or K-fold CV methods from the results of simulations and real data study.

The proposed 1-fold CV is useful when a statistical testing is conducted for validation. However, one may want to calculate an unbiased performance parameter, such as AUC of an ROC curve for a binary outcome and Herrall's c-index (Harrell, et al. [22]), for a survival outcome. To this end,

we need a validation set independent of a training set, so that hold-out or other CV methods should be used. The 1-fold CV will always give an over-fitting estimates of performance parameter since it uses whole data set for both training and validation.

One may think that our 1-fold cross validation (CV) is similar to LOOCV. For a data set with sample size n, the size of training set is $n - 1$ for LOOCV, while that of 1-fold CV is n. In that sense, the efficiency of training will be similar between the two methods. However, the way each data point is used for validation is very different between the two CV methods. In LOOCV, if a data point is used for training, it is not used only for validation, whereas in 1-fold CV, each data point is used for both training and validation at the same time. So, the validation power of LOOCV may be slightly higher than 1-fold CV. However, the prediction efficiency will be determined by the efficiency of both training and validation. From our simulation studies, we find that 1-fold CV has higher prediction efficiency than existing methods including LOOCV. Furthermore, in training, 1-fold CV fits only one model, while LOOCV fits n prediction models with a similar data size (n vs. $n - 1$). Note that the model fitting procedure is repeated during CV also, so that LOOCV takes about n times longer computing time of that of 1-fold CV for the whole prediction procedure. The difference in computing time between the two methods will be prominent with a large sample size, number of features, and replications for CV.

Computer programs developed for data analysis of this paper are available from the authors upon request.

References

[1] Mayer DG, Butler DG. Statistical Validation. Ecol Modell. 1993;68:21-32.

[2] Stone M. Cross-Validatory Choice and Assessment of Statistical Predictions. J R Stat Soc B (Methodol). 1974;36:111-133.

[3] Stone M. An Asymptotic Equivalence of Choice of Model by Cross-Validation and Akaike's Criterion. J R Stat Soc B (Methodol). 1977;39:44-47.

[4] Allen DM. The Relationship Between Variable Selection and Data Agumentation and a Method for Prediction. Technometrics. 1974;16:125-127.

[5] Khera R, Haimovich J, Hurley NC, McNamara R, Spertus JA, et al. Use of Machine Learning Models to Predict Death After Acute Myocardial Infarction. JAMA Cardiol. 2021;6:633-641.

[6] Christodoulou E, Ma J, Collins GS, Steyerberg EW, Verbakel JY, et al. A Systematic Review Shows No Performance Benefit of Machine Learning Over Logistic Regression for Clinical Prediction Models. J Clin Epidemiol. 2019;110:12-22.

[7] Diana G, Tommasi C. Cross-Validation Methods in Principal Component Analysis: A Comparison. Stat Methods Appl. 2002;11:71-82.

[8] Xu Y, Goodacre R. On Splitting Training and Validation Set: A Comparative Study of Cross-Validation,Bootstrap and Systematic Sampling for Estimating the Generalization Performance of Supervised Learning. J Anal Test. 2018;2:249-262.

[9] Kim JH. Estimating Classification Error Rate: Repeated Cross-Validation, Repeated Hold-Out and Bootstrap. Computational statistics and data analysis. 2009;53:3735-3745.

[10] Borra S, Di Ciaccio AD. Measuring the Prediction Error. A Comparison of Cross-Validation, Bootstrap and Covariance Penalty Methods. Comp Stat Data Anal. 2010;54:2976-2989.

[11] Isler Y, Narin A, Ozer M. Comparison of the Effects of Cross-Validation Methods on Determining Performances of Classifiers Used in Diagnosing Congestive Heart Failure. Meas Sci Rev. 2015;15:196-201.

[12] Pang H, Jung SH. Sample Size Considerations of Prediction-Validation Methods in High-Dimensional Data for Survival Outcomes. Genet Epidemiol. 2013;37:276-282.

[13] Liu L, Gao J, Jung SH, Beasley G. LASSO and Elastic Net Tend to Over-Select Features. Mathematics. 2023;11:3738.

[14] Simon RM, Subramanian J, Li MC, Menezes S. Using Cross-Validation to Evaluate Predictive Accuracy of Survival Risk Classifiers Based on High-Dimensional Data. Brief Bioinform. 2011;12:203-214.

[15] Tolles J, Meurer WJ. Logistic Regression: Relating Patient Characteristics to Outcomes. JAMA. 2016;316:533-534.

[16] Cox DR. Regression Models and Life-Tables. J R Stat Soc B (Methodol). 1972;34:187-202.

[17] Hanley JA, McNeil BJ. The Meaning and Use of the Area Under a Receiver Operating Characteristic (Roc) Curve. Radiology. 1982;143:29-36.

[18] Mason SJ, Graham NE. Areas Beneath the Relative Operating Characteristics (Roc) and Relative Operating Levels (Rol) Curves: Statistical Significance and Interpretation. Q J R Meteorol Soc. 2002;128:2145-2166.

[19] Mann HB, Whitney DR. On a Test of Whether One of Two Random Variables Is Stochastically Larger Than the Other. Ann Math Statist. 1947;18:50-60.

[20] Peto R, Peto J. Asymptotically Efficient Rank Invariant Test Procedures. J R Stat Soc A. 1972;135:185-207.

[21] Farrow NE, Holl EK, Jung J, Gao J, Jung SH, et al. Characterization of Sentinel Lymph Node Immune Signatures and Implications for Risk Stratification for Adjuvant Therapy in Melanoma. Ann Surg Oncol. 2021;28:3501-3510.

[22] Harrell FE, Califf RM, Pryor DB, Lee KL, Rosati RA, et al. Evaluating the Yield of Medical Tests. JAMA. 1982;247:2543-2546.