

Highlights

Profile Electoral College Cross-Validation

Zishu Zhan, Yuhong Yang

- Regular k -fold cross-validation (CV) has three major weaknesses
- An electoral-college-style voting CV provides more reliable info on the candidate procedures
- Multiple data splitting ratios in CV yield a profile of performances of the candidates
- The new profile electoral-college CV selects the best candidate with high probability, while offering valuable insight unavailable in previous CV methods

Profile Electoral College Cross-Validation

Zishu Zhan^a, Yuhong Yang^{b,*}

^aSchool of Statistics, Remin University of China, Beijing 1000872, China

^bSchool of Statistics, University of Minnesota, Minneapolis, MN 55455, USA

ARTICLE INFO

Keywords:

cross-validation
model selection
data splitting ratio
reverse k -fold
cross-validation paradox

ABSTRACT

Cross-validation (CV), while being extensively used for model selection, may have three major weaknesses. The regular 10-fold CV, for instance, is often unstable in its choice of the best model among the candidates. Secondly, the CV outcome of singling out one candidate based on the total prediction errors over the different folds does not convey any sensible information on how much one can trust the apparent winner. Lastly, when only one data splitting ratio is considered, regardless of its choice, it may work very poorly for some situations. In this work, to address these shortcomings, we propose a new averaging-voting based version of cross-validation for better comparison results. Simulations and real data are used to illustrate the superiority of the new approach over traditional CV methods.

1. Introduction


Cross-validation (CV) is one of the most widely used strategies for model/procedure selection. Basically, part of the data is used for training each competing model or procedure, and the rest of the data is used to assess their performances. The process is repeated multiple times and the candidate with the best overall performance is chosen.

Given a data splitting ratio (DSR), both exhaustive data splitting (i.e., considering all possible data splittings at the ratio) [1, 26] and partial data splitting [14] have been studied. The latter saves the computation cost and includes k -fold CV [8], balanced incomplete CV [24], and repeated learning-testing (RLT) [8, 9, 34]. See [3] for a comprehensive and informative review on the general topic of cross-validation.

CV has been extensively applied and theoretically examined in many contexts. The two main aspects linked to the adequacy of a learning model, bias and variance, have been investigated in detail (see, e.g., [6, 20, 22, 16]). However, there are closely related but subtly different goals of using CV, and a lack of proper differentiation may have contributed to several widely seen misconceptions on CV [35]. In particular, accurate estimation of the prediction errors is not necessarily aligned with the goal of finding out the best candidate, which leads to the “cross-validation paradox” that better training (with more data) and better evaluation (with more data) actually can significantly worsen the wrong selection probability [30, 35, 12]. A bottom line is that when the goal is to identify the best candidate, the evaluation data proportion has to be large enough, which offers two-fold benefits, namely, 1) obviously there are more data points to evaluate the candidate models/procedures, a *direct benefit*; 2) the reduced sample size of training, as long as it does not tip over the ranking of the candidates, may magnify the differences of the close competitors to bless the evaluation step, an *indirect benefit*. In this work, we will focus on the task of using CV to pick the best competitor, and the other goals of CV (e.g., [35, 4]) will not be much addressed.

In this paper, we argue that the current standard practice of CV can be much improved by bringing in two important aspects. One is that the current CV approach does not present a proper quantification of how different the competitors are. Note that heuristic standard errors of CV in the literature, such as used in one-standard-error rule for tuning parameter selection, that depend on variance estimation based on multiple CV errors from different data splittings, are actually not valid due to obvious dependence of the CV errors from different folds/data splittings (see, e.g., [30]). As shown in [6], there does not exist generally applicable unbiased variance estimator of the CV prediction error. When the candidate learning procedures are highly adaptive and data-driven, the task of estimating the variability of CV prediction error becomes really complicated – so much so that there has not been any reasonably general success, to the best of our knowledge.

*Corresponding author

 yangx374@umn.edu (Y. Yang)

 <http://www.stat.umn.edu/~yyang> (Y. Yang)

ORCID(s): 0000-0003-3618-3083 (Y. Yang)

As a solution, we propose the use of voting in multiple data splittings to offer insight on how close the competitors are. In theory, with a large sample, the best candidate should have winning fraction approaching 1. For the specific data, with the limited information, the winning frequencies of the competitors properly describe how distinct the performances of the candidates are relative to the sample size, and this quantification naturally sheds light on how much to trust the comparison of the candidates at the given DSR.

The other important aspect is that no single DSR is generally sufficient for a reliable CV comparison. For the purpose of estimation (instead of model identification), in the context of density estimation using least squares projection estimators into linear subspaces, Arlot and Lerasle [4] derived non-asymptotic oracle inequalities for k -fold CV, which supports that 5-fold or 10-fold may be good choices for optimal density estimation. For our goal of model identification, we propose the use of a profile of voting frequencies at multiple DSRs to reach a much more reliable and insightful conclusion on who is the best candidate. Also, we integrate the averaging of the prediction errors (as is done in regular CV) into the voting system for more effective/stable results. With these distinctive features, our new version is called profile electoral college CV, or PEC-CV in short. Furthermore, the same strategy may be applicable to other modified CV methods (see, e.g., [27] for covariate shift adaptation, [29] for selection of the number of clusters, [33] for penalized high dimensional linear model and [19] for kernel-based algorithms) by properly modifying the criterion function.

Clearly, our proposed new CV strategy is computationally more demanding due to the use of multiple DSRs. For applications where the CV is used to quickly choose one reasonable (instead of the best) candidate in a time-sensitive manner, it is perhaps enough or even preferred to use, e.g., regular 10-fold CV. However, for applications where insight and interpretation are sought based on the selected procedure, the extra computation cost may be worthwhile. The use of our profile CV may much improve *stability* and *predictability* of the outcome, as will be demonstrated later in this paper. When variable selection methods are compared by CV, for instance, the profile CV may also improve *interpretability* of the variable selection outcome. Note that *stability*, *predictability* and *interpretability* are key principles of learning from data [32].

In this paper, we will focus on regression with a continuous response, but the methodology works more generally for classification and other generalized linear modeling frameworks where the accuracy of the prediction of the response can be properly assessed via a loss function. An example of classification will be given to illustrate this point.

The rest of the paper is organized as follows. In Section 2, we highlight several weaknesses of the commonly used k -fold CV, as represented by the regular 10-fold CV. The new EC-CV procedures are defined in Section 3. Then, the construction and regular patterns of PEC-CV for model selection are presented in Section 4. Properties of the new CV methods are stated in Section 5. In Section 6, an illustration based on real data is given. Concluding remarks are in Section 7. The proof of the theorem is in the appendix. Some details for the numerical work and additional supporting materials are provided in a supplementary file.

2. Problems with the regular 10-fold CV

In this section, with illuminating simulation examples, we illustrate several major weaknesses of a regular k -fold CV. Since 10-fold CV is widely suggested in the literature, it will be used as a representative, although other choices will be considered as well. The standard k -fold CV starts with a partitioning of the data into k sub-samples of (roughly) equal size. Each of these sub-samples in turn plays the role of evaluation/assessment sample, while the rest are used to train the candidate procedures. For each candidate procedure, the k assessment results from the folds are then totaled to produce a single performance quantification. The procedure with the best performance is selected. In the literature, 10-fold CV is the favorite to use [17] (see, e.g., [35] for cautionary views).

We identify three weaknesses of the regular k -fold CV.

2.1. Instability of the regular 10-fold CV

The instability of regular k -fold CV has been pointed out in the literature (see, e.g., [22, 35, 16]) and a repeated k -fold CV can alleviate the problem. However, the message needs to be emphasized more from the model selection perspective (in addition to prediction error estimation), with strong examples.

With the data given, prediction performance estimate by the regular 10-fold CV has uncertainty due to the randomness of partitioning the sample into ten folds. It turns out that in many situations, this uncertainty is large enough to make the final selection result unnecessarily volatile. It often can easily be biased by cherry-picking a specific data

Table 1
CV selection of the better model

	Proportion
5 ⁻¹ -fold	0.93
Repeated 5 ⁻¹ -fold	1.00
2-fold	0.56
Repeated 5-fold	1.00
3-fold	0.57
Repeated 2-fold	1.00
5-fold	0.69
Repeated 5-fold	1.00
10-fold	0.77
Repeated 10-fold	1.00

79 splitting to possibly favor a candidate model/procedure.

80

81 **Example 1:** Here the data, with sample size $n = 100$, are generated from a linear model, and the true model is
 82 compared with an overfitting model. In this example, five k -fold CV are considered: 2-fold, 3-fold, 5-fold, 10-fold and
 83 the reverse 5-fold, denoted as 5⁻¹-fold, which means that each time one fold is used for training and the rest 4 folds
 84 are used for evaluation. The repeated CV method conducted here averages the prediction errors over r repetitions of
 85 the regular k -fold CV for some r to be described below.

86 When the k -fold CV is implemented with one random data splitting (as is usually done in most practices), among
 87 100 replications of this, the fraction of selecting the true model is 0.93, 0.56, 0.57, 0.69, 0.77 for $k = 5^{-1}, 2, 3, 5, 10$
 88 respectively. In contrast, when the k -fold CV is repeated (with random data partitions) 48 times, 120 times, 80 times,
 89 48 times, 24 times, respectively for $k = 5^{-1}, 2, 3, 5, 10$ (so that the total number of training-evaluation steps is the same
 90 for all of them for comparability), the frequency of selecting the true model based on the winning frequencies of the
 91 competitors over the data splittings are all substantially increased to 1. This example clearly shows in Table 1 that the
 92 k -fold CV, when implemented only once, can be quite unstable, and the repeated k -fold CV with voting substantially
 93 improves in such cases. Note that the fact that 5⁻¹-fold performs better than 10-fold is not surprising in this case, which
 94 is in line with the cross-validation paradox (see, [24, 30]). The details of the example are in the supplementary file.

95 2.2. The regular 10-fold CV is not informative

96 Another problem related to the 10-fold CV is that the summing up over ten folds gives only one number by av-
 97 eraging the results. But it is unclear if the CV winner has a decisive edge over the others or not, and one does not
 98 have a good sense of how reliable the comparison is. As mentioned already, the commonly seen "standard errors" of
 99 CV errors are actually unreliable and thus cannot be used to address these issues. We illustrate this point by an example.

100

101 **Example 2:** In this example, (detailed are in the supplementary file), we compare a true model with a wrong model
 102 in two situations. In the first situation, the wrong model is severely wrong, but in the second situation, the wrong model
 103 is just slightly wrong. Two data sets were generated from the true model respectively and the 10-fold CV is used to
 104 choose between the true and the wrong models. It turns out that in both cases, the true model is selected. The standard
 105 application of CV would stop here, declaring the true model to be the winner. There is no hint on how much confidence
 106 we have in the claim and if the winner has won the competition easily.

107 In contrast, for the to-be-proposed EC-CV method, it gives the frequency of winning of the true model over the
 108 wrong model by CV over multiple random data splittings. The results are presented in Table 2 based on $r = 100$ data
 109 splittings. The table is very informative and shows a drastic contrast between the two situations: In the first situation,
 110 the true model is clearly much better (at least for the DSR), but in the second, it is still better, but not dominantly so.
 111 Thus the new EC-CV provides crucial info on the comparison of the candidate procedures.

112 To illustrate the usefulness of the info brought up by the EC-CV, we replicate the above experiment $N = 100$ times
 113 and report the number of times that the true model wins by the regular 10-fold CV in Table 3. It clearly shows that in
 114 the first situation, the true model was selected every time, but in the second, the true model was incorrectly declared
 115 worse 29 times. Therefore the winning frequency of EC-CV offers a proper measure on how comfortably the winner
 116 tops the other candidate models/procedures.

Table 2

The winning frequency of repeated 10-fold CV in model selection: Severely wrong v.s. Less wrong

Severely wrong		Less wrong	
true model	wrong model	true model	wrong model
100	0	65	35

Table 3

The results of 10-fold CV in model selection based on $N = 100$ replicated data generating process: Severely wrong v.s. Less wrong

Severely wrong		Less wrong	
true model	wrong model	true model	wrong model
100	0	71	29

2.3. No DSR can be the best generally

There are different recommendations on the DSR for CV [24, 17, 2, 22, 4]. Yang [30, 31] and Zhang and Yang [35] have provided a theoretical understanding that the choice of DSR for CV needs to care for two possibly conflicting directions for selection consistency: More observations for evaluation are needed to distinguish the close competitors; At the same time, the training size cannot be too small so as to avoid a possible change of the relative ranking of the candidate procedures. These results suggest that the optimal DSR is associated with the nature of the target regression function, the noise level, the sample size, and the natures of the candidate estimators.

In one direction of the spectrum of optimal DSR, we have the cross-validation paradox mentioned in [30] that pushes for most observations for the evaluation. Suppose a specific data splitting (e.g., 10-fold) scheme does a good job in comparing two slightly different good parametric models. Now suppose we have more independent and identically distributed data, and we maintain the splitting ratio. With the improved estimation capability, we expect better results in comparing the two models. In reality, surprisingly, the probability of selecting the better model is decreased. In contrast, if we increase the proportion of the evaluation part as the sample size increases, the CV comparison of the two procedures works better and better, approaching the perfect decision. When comparing models that are close to each other, the large size of observations for the evaluation set smoothes out the fluctuations of the CV errors and increases the capability to distinguish the close competitors. On the contrary, increasing training size narrows the estimation accuracy difference between the close competitors and makes the procedures more difficult to be distinguished.

In the other direction of the spectrum of optimal DSR, for instance, in the context of comparing a parametric estimator and a kernel estimator, it is actually better to have more observations (but not too many) for training. The reason is that the two estimators behave very differently, having easily distinguishable rates of convergence, and the nonparametric estimator typically relies on a large training size to be effective, showing its advantage. Too small a training size would lead to misleading performance of the kernel method when it is actually much better at the full sample size.

From the above, there is no magic single data splitting approach that works generally. An illustration is given below.

Example 3: There are two scenarios here, both with $n = 120$. In Scenario 3.1, like Example 1, a true model and an overfitting model are considered as the candidates. In Scenario 3.2, data are generated from another model. Random Forest (RF) [7] and Support Vector Machine (SVM) [11] are compared. Here EC-CV at 5⁻¹-fold, 2-fold, 3-fold, 5-fold and 10-fold are used. In addition, leave-one-out (LOO), which is an extreme case of k -fold CV with only one observation for evaluation, is included for comparison.

Table 4 presents the simulated probabilities of selecting the better estimator (the one with smaller prediction error given the data, see Section 4.1 for a precise definition) based on $N = 100$ replications. Apparently, 5⁻¹-fold EC-CV is superior to the other DSRs in Scenario 3.1 but is inferior in Scenario 3.2. Note also that LOO may perform very poorly for comparing different models/procedures due to its use of a single observation in evaluating each model in every data splitting. The simulation clearly demonstrates that even with the strengthened EC-CV, it is not good to indiscriminately recommend one DSR, such as 10-fold. We note that most of the differences between the selection probabilities of the CV methods in Table 4 are statistically significant at level 0.05. Details on this and other aspects of the example are given in the supplementary file.

Table 4

Selections of the better procedure by EC-CV at 5 fold choices and LOO based on $N = 100$ replicated data generating process

DSR	Scenario 3.1	Scenario 3.2
5 ⁻¹ -fold	0.90	0.02
2-fold	0.70	0.56
3-fold	0.71	0.81
5-fold	0.69	0.88
10-fold	0.73	0.90
LOO	0.42	0.86

155

156 3. The electoral college CV

157 We here propose a new CV method called electoral college cross-validation (EC-CV). Its construction intends to
 158 address the first two weaknesses of the regular k -fold CV as stated in the previous section (the third weakness will
 159 be dealt with in the next section). First, some repetitions of the random data splitting are needed to stabilize the
 160 selection result. Second, a repeated k -fold simply by averaging the prediction errors over the repeated random data
 161 splittings would not address the second weakness. Our solution is to stabilize a regular k -fold by voting based on
 162 a number of repeated k -fold comparisons. Specifically, for each random data splitting, the candidate with the best
 163 overall performance based on the k -fold CV receives one vote. After a number of random data splittings, the candidate
 164 that receives the most votes is declared the final winner. This way, the winning fraction of the winner conveys a clear
 165 sense of competitiveness of the candidate models. The winning fraction of nearly 1 presents a totally different picture
 166 compared to the winning fraction of 55% for instance.

167 This kind of CV is, in some spirit, similar to the electoral college voting system of the United States. For each state
 168 (analogous to each data splitting), after the counts in the different precincts (analogous to different folds) are totaled,
 169 the resulting state-wise winner receives the electoral votes; and the state votes (analogous to winning frequencies of
 170 the candidate procedures over the data splittings) are tallied to decide the overall winner. Hence the name EC-CV.
 171 In summary, in EC-CV, we first adopt the k -fold CV process with averaging (or totaling) over the k -folds, and then
 172 sum up the results of r data splittings by voting. We will use the notation EC-CV(r) to indicate the number of data
 173 splittings.

174 There can be other ways to do a repeated CV. Some obvious choices are: 1. With r data splittings, there are $r \times k$
 175 prediction errors for each candidate, and we can do averaging over the $r \times k$ prediction errors; 2. Instead of averaging
 176 over the k -folds, for each data splitting, we can decide the winner based on frequencies of minimizing the prediction
 177 errors over the k -folds, and then vote based on the multiple data splittings; 3. We can do voting directly based on the
 178 $r \times k$ comparisons; 4. RLT is a kind of CV that several subsets are chosen randomly and independently from the data
 179 [8]. However, RLT may be inefficient due to pure randomness of selected observations in the training samples. Here
 180 we also consider a version of voting based RLT: The data are randomly put into training and testing at the DSR and the
 181 candidate with smaller prediction error on the test data receives one vote; After a number of repetitions of this process,
 182 the candidate with most votes is the final winner. These alternative versions of CV will be referred to as *Averaging*,
 183 *Voting-Voting*, *Voting*, *Voting RLT*, respectively.

184 Generally speaking, our EC-CV is typically more efficient than the *Voting-Voting*, *Voting* and *Voting RLT* versions
 185 of CV because the averaging-voting has a desirable stabilizing effect with the first averaging step. The pure *Averaging*
 186 may in fact perform significantly better than the pure voting-based methods (*Voting-Voting*, *Voting* and *Voting RLT*),
 187 but it may also lose substantially to the voting methods sometimes as will be seen in Table 5. Overall, the EC-CV that
 188 combines averaging and voting seems to be the best performer. Again, in contrast to EC-CV, *Averaging* (i.e., averaging
 189 over both folds and data splittings) would not offer a sensible measure on reliability of the CV selection outcome.

190 We illustrate the aforementioned advantage of EC-CV over the pure voting based CV methods as well as the simple
 191 repeated CV in the following example.

192

193 **Example 4:** There are two scenarios here. In Scenario 4.1, the data, with $n = 120$, are generated the same way
 194 as in Scenario 3.1 and the two candidate procedures to be compared stay the same as well. In Scenario 4.2, data with

Table 5
CV selection proportion of the better procedure

	DSR	Averaging-Voting (EC)	Voting-Voting	Voting	Voting RLT	Averaging
Scenario 4.1	5 ⁻¹ -fold	0.90	0.84	0.82	0.80*	0.85
	2-fold	0.77	0.63*	0.62*	0.57*	0.75
	3-fold	0.73	0.54*	0.55*	0.50*	0.74
	5-fold	0.74	0.40*	0.45*	0.48*	0.71
	10-fold	0.71	0.34*	0.36*	0.38*	0.73
Scenario 4.2	5 ⁻¹ -fold	0.84	0.81	0.81	0.80	0.78
	2-fold	0.82	0.82	0.83	0.80	0.68*
	3-fold	0.80	0.81	0.79	0.76	0.69*
	5-fold	0.78	0.72	0.72	0.74	0.67*
	10-fold	0.77	0.75	0.76	0.78	0.72

195 $n = 120$ are generated from a linear model with 10 outliers in response, and linear regression with ordinary least
 196 squares is compared with least squares regression after excluding the outliers in training data (see the supplementary
 197 file for details).

198 We compare EC-CV, the three versions of voting based CV methods with $k = 5^{-1}, 2, 3, 5, 10$ and the Aver-
 199 aging CV (note that for the Voting RLT, delete- n_e with $n_e = 0.8n, 0.5n, 0.33n, 0.2n, 0.1n$ matches the k -fold with
 200 $k = 5^{-1}, 2, 3, 5, 10$, respectively). Given the data, for the comparison to be fairer, 240 data splittings are done for the
 201 Voting RLT, $240/k$ (or $240k$ for $k = 5^{-1}$) for the other methods so that they all have the same number of procedure
 202 training and prediction evaluations. In particular, EC-CV(48), EC-CV(120), EC-CV(80), EC-CV(48), and EC-CV(24)
 203 are applied under $k = 5^{-1}, 2, 3, 5, 10$, respectively. We replicate the data generating process $N = 100$ times, and the
 204 proportions that the conditionally better procedure (see Section 4.1 for a precise definition) wins are of interest. The
 205 results are shown in Table 5. Note that the symbol * indicates that EC-CV produces significantly better selection
 206 than the other method based on one-sided paired t -test ($\alpha = 0.05$). The proposed EC-CV (Averaging-Voting CV)
 207 outperforms the other CV methods in an overall sense.

208 With the data being randomly generated, EC-CV can occasionally lead to an unwarranted support to a procedure
 209 (perhaps similar to the EC voting system in the real election?). Later, theoretical results will show that under proper
 210 conditions, EC-CV does select the better candidate with probability going to one.

211 4. The profile EC-CV

212 Now we address the third weakness of the regular k -fold CV stated in Section 2 that a single choice of k cannot
 213 be generally suitable for procedure comparisons. A natural approach is to consider multiple DSRs and obtain a profile
 214 of competitiveness of the candidate regression procedures at different DSRs. This profile provides rich information
 215 on the relative performances of the candidates and reliability of the final selection. We call this profile EC-CV, or
 216 PEC-CV.

217 Specifically, we may consider 5⁻¹-fold, 4⁻¹-fold, 3⁻¹-fold, 2-fold, 3-fold, 4-fold, 5-fold, 6-fold, 10-fold and 12-fold
 218 for PEC-CV. In practice, to save the computational cost, 5⁻¹-fold, 2-fold, 4-fold, and 10-fold are usually good enough
 219 to delineate the profile to make a sound decision.
 220

221 4.1. Two notions to compare the candidate procedures

222 In the earlier examples of comparing different models/procedures, one candidate is based on the true model or the
 223 other is clearly inferior (see the supplementary file for details). In a general situation, care is needed to define which
 224 candidate is the best. There are two valid notions that can be used and we formally define them below.

225 Let δ_1 and δ_2 be two procedures to estimate a target regression function f . Based on data $\mathcal{D}^n = (X_i, Y_i)_{i=1}^n$ with n
 226 iid observations drawn from a distribution μ , δ_1 and δ_2 produce estimators $\hat{f}_{n,1}$ and $\hat{f}_{n,2}$, respectively. Let (X, Y) be
 227 an independent copy from μ . Let $L(Y, \hat{f}(X))$ denote the prediction loss of interest when using an estimator $\hat{f}(X)$ to
 228 predict Y at given X . We will focus on the square loss: $L(Y, \hat{f}(X)) = (Y - \hat{f}(X))^2$. Consider a CV method γ that
 229 chooses between δ_1 and δ_2 based on \mathcal{D}^n .
 230

231 **Definition:** Let δ_1 and δ_2 be the procedures to be compared.

1. Given the observed data D^n , δ_1 is said to be conditionally better than δ_2 if

$$E_{(X,Y)\sim\mu} L(Y, \hat{f}_{n,1}(X)) < E_{(X,Y)\sim\mu} L(Y, \hat{f}_{n,2}(X)),$$

232 where $E_{(X,Y)\sim\mu}$ denotes expectation taken with respect to the randomness of (X, Y) drawn from μ independently
233 of the data D^n .

2. For the data generating distribution μ , δ_1 is said to be (unconditionally) better than δ_2 if

$$E_{D^n, (X,Y)\sim\mu} L(Y, \hat{f}_{n,1}(X)) < E_{D^n, (X,Y)\sim\mu} L(Y, \hat{f}_{n,2}(X)),$$

234 where $E_{D^n, (X,Y)\sim\mu}$ denotes expectation taken with respect to the randomness of both D^n and (X, Y) indepen-
235 dently drawn from μ .

236 From the definition, the two notions focus on behaviors of the candidate procedures for the present realized data or
237 in repeated applications with data generated from the specified distribution. Ideally, we want to select the conditionally
238 better procedure, but it is a more challenging goal to achieve than selecting the unconditionally better procedure.

239 Now for a CV method γ , its performance can be naturally measured in terms of its probability of selecting the
240 conditionally or unconditionally better procedure between δ_1 and δ_2 . Since these probabilities under the two notions
241 above are typically analytically intractable, one relies on simulations for their calculations. Specifically, we draw M
242 (large, say 10000) iid observations, $(X_m, Y_m)_{m=1}^M$ from μ independently of the data D^n . Then we obtain Monte Carlo
243 approximations of $E_{(X,Y)\sim\mu} L(Y, \hat{f}_{n,j}(X))$ by $1/M \sum_{m=1}^M L(Y_m, \hat{f}_{n,j}(X_m))$, $j = 1, 2$. Consequently the conditionally
244 better procedure is determined.

245 To find which procedure is unconditionally better, we replicate the data generation process of D^n a large number
246 of times (say 100 or 1000) and obtain the average losses of the two procedures as Monte Carlo approximations to
247 $E_{D^n, (X,Y)\sim\mu} L(Y, \hat{f}_{n,j}(X))$, $j = 1, 2$, and decide accordingly. Note that for regression, the square loss is usually used,
248 but for classification, the 0-1 loss is more appropriate. The above approach is applied when comparing the candidate
249 procedures in the numerical studies in this work. More specifically, in each simulation setting, probabilities of choosing
250 the conditionally or unconditionally better model/procedure are calculated for a CV method: The data are generated
251 a large number of times and the proportion that the CV method chooses the conditionally better procedure and that it
252 chooses the unconditionally better procedure are returned.

253 4.2. Common patterns of PEC-CV

254 We have done a number of simulations comparing various regression procedures and have summarized the fol-
255 lowing frequently occurring and representative patterns. Several figures will be presented, which are obtained for
256 simulated data and for each k , $240/k$ random data splittings are considered for calculating the voting frequencies of
257 the candidate procedures. The frequencies of having *smaller* prediction errors are plotted in the figures. Details of
258 the data generation and the candidate procedures can be found in the supplementary file. We should point out that
259 these patterns are not meant to be exhaustive and there certainly can be more. Note that in all the following examples,
260 method 1 (solid line with purple color) is truly conditionally better than method 2.

261 • Pattern 1 (*Dominating*). As shown in Fig. 1 (as an extreme case), one method wins the competition decisively
262 at all DSRs. This happens when one candidate is just superior to the other at sample sizes reasonably close to
263 that of the data at hand. The PEC-CV plot correctly endorses it with really high confidence.

264
265 • Pattern 2 (*Indifferent*). As illustrated in Fig. 2, the gaps between the winning frequencies of the two methods are
266 rather small at the different DSRs, and can be both positive and negative. Here method 2 has a slightly higher
267 winning frequency at DSR 11:1, but as we have explained before, when the evaluation size is small, a marginally
268 higher winning frequency is not a piece of strong evidence in support of the method. This profile indicates the
269 candidates perform very similarly, and one should not attach much confidence on the final selection between the
270 two.

271

Profile Electoral College Cross-Validation

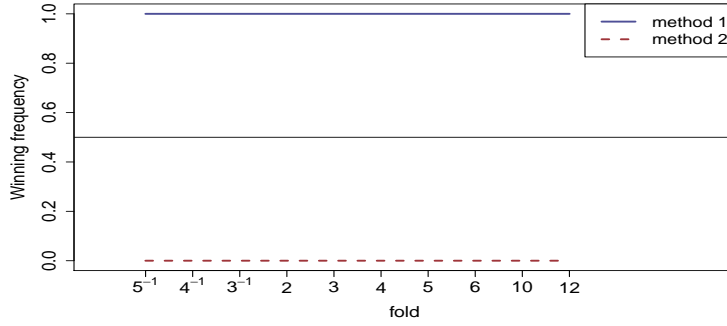


Figure 1: Common pattern 1 of PEC-CV: Dominating

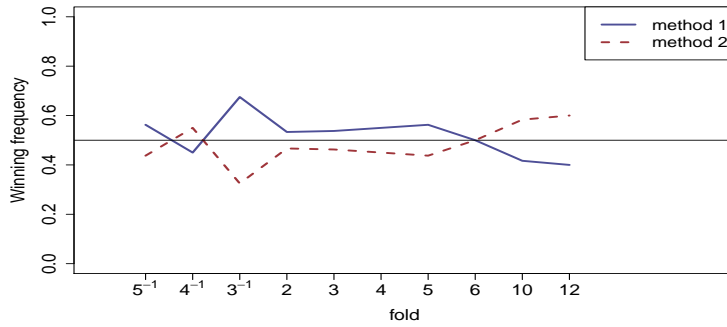


Figure 2: Common pattern 2 of PEC-CV: Indifferent

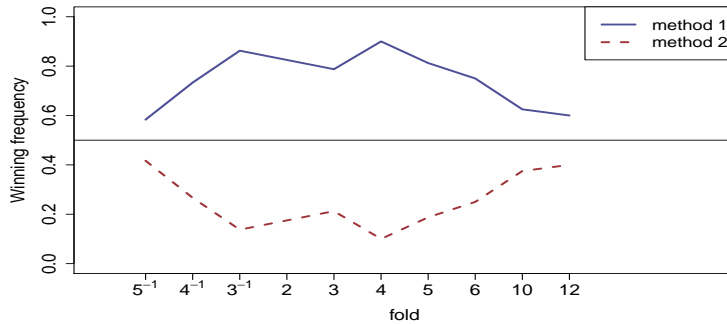


Figure 3: Common pattern 3 of PEC-CV: Marginally better

- 272 • Pattern 3 (*Marginally better*). As shown in Fig. 3, consistently at all the different DSRs, one method is clearly

273 above the 50% line, but to a limited degree, especially towards the end. This profile suggests that we are quite

274 confident that one method is generally better, but only marginally so.

275
- 276 • Pattern 4 (*Strong switching*). This is illustrated in Fig. 4. Here, the winning frequency of one method (denoted

277 as method 1) is below 50% in the beginning but increases substantially (passing 50% line eventually) as the

Profile Electoral College Cross-Validation

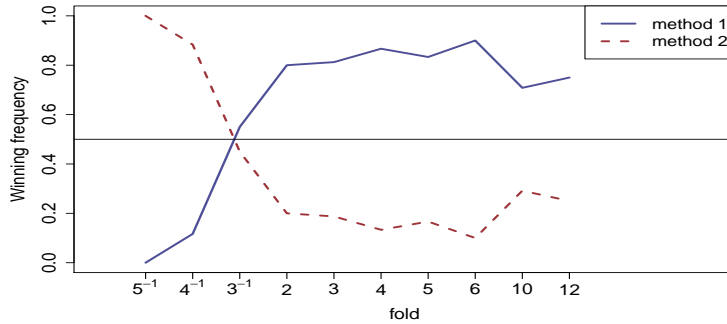


Figure 4: Common pattern 4 of PEC-CV: Strong switching

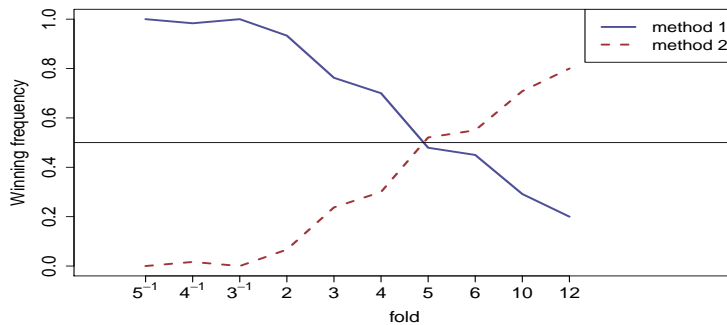


Figure 5: Common pattern 5 of PEC-CV: Marginal switching

278 training sample size increases. The switching can happen early or late, but the winning frequency of method
 279 1 is close to 1 towards the end. This situation may happen when the estimation process of method 1 is more
 280 complex than method 2, and therefore, method 1 suffers from the low sample size more than method 2. When
 281 the training size gets large enough, method 1 becomes clearly more competitive than method 2. This profile
 282 shows that the selection of method 1 at the full sample size is a confident one (but method 2 may be preferred
 283 at a smaller sample size).
 284

285 • Pattern 5 (*Marginal switching*). As depicted in Fig. 5, method 2 has transitioned to be better as the number of
 286 folds increases, but its advantage over method 1 is not quite certain. Here when k gets larger, the number of ob-
 287 servations used to assess the predictive performance becomes smaller, making the comparison of the candidates
 288 less certain for a tough situation (which is the reason behind the cross-validation paradox). For this delicate
 289 case, if we just use the regular 10-fold CV, the selection outcome is quite random and unreliable, and we have to
 290 handle it with kid gloves. The profile CV provides a bigger picture and it suggests that method 2 might be better
 291 than method 1 with more training samples, but the confidence level on this is low. Since method 1 is actually
 292 better, the decision to choose method 2 by focusing only on 10-fold or 12-fold would be wrong. The PEC-CV
 293 profile can clearly warn against simply trusting the selection outcome by the popular 10-fold. Furthermore, the
 294 PEC-CV may actually prefer method 1 as the winner when integrating together the performances at the different
 295 DSRs (see Section 4.3).
 296

297 • Pattern 6 (*V-shape*). As shown in Fig. 6, method 1 has winning frequencies above 50%, but it roughly has a

298
299
300
301
302
303
304
305
306

V-shape: the winning frequency of method 1 is high at both ends but lower in the middle. The profile suggests that the choice of method 1 is most likely safe at the full sample size. We note that sometimes it may be possible that the winning frequency of method 1 can drop close to or even slightly below 0.5 in the middle. Note that such a profile can occur when comparing model selection methods. For example, when using BIC [23] or Least Absolute Shrinkage and Selection Operator (LASSO) [28] to select among linear regression models, when the true coefficients have different magnitudes, it has been observed in the literature that when the training sample size is small, LASSO performs better, but when the sample size gets larger, BIC performs better, and when the sample size gets much larger, LASSO wins back the competition [18].

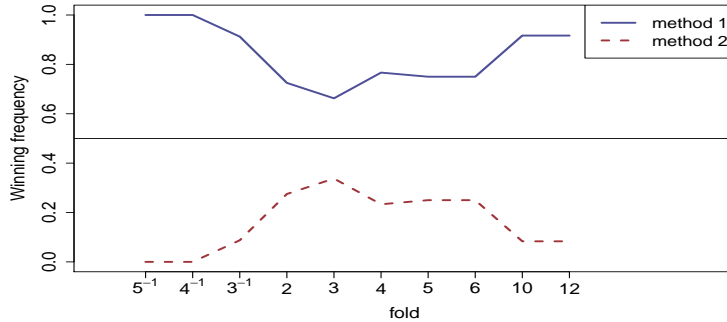


Figure 6: Common pattern 6 of PEC-CV: V-shape

307
308
309
310
311
312
313

- Pattern 7 (*Asymmetric*). This pattern is unique in some sense. Sometimes, the two competing methods may give identical regression estimates. For instance, in the comparison of BIC and BICc [15, 35], they may actually select the same model, in which case no one wins the competition. Thus their winning frequencies do not necessarily add to 1 and it is no longer the case that one wanes while the other waxes. As shown in Fig. 7, the winning frequency curves are asymmetrical about the line of 0.5. The profile plot suggests that the two methods agree frequently, but it is quite clear in this case that method 1 should be selected.

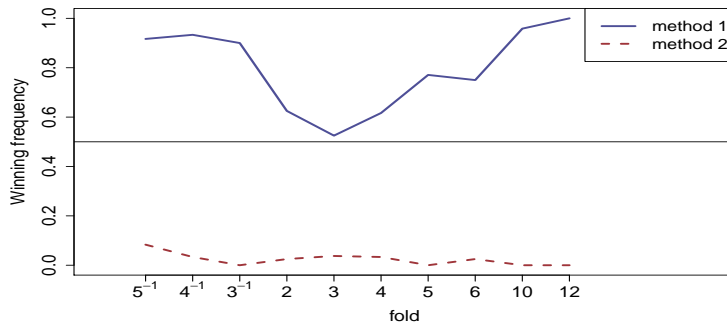


Figure 7: Common pattern 7 of PEC-CV: Asymmetric

314
315
316

As explained above, the PEC-CV plots provide much richer and more reliable info on relative performances of the competitors compared to the regular CV. We have highlighted several common patterns, based on which more robust and accurate decisions can be made. Of course, due to randomness of the data, sometimes the PEC-CV plot may falsely

317 recommend a candidate. But, very importantly, in most such cases, we would humbly admit we are not confident about
 318 the selection outcome when the PEC-CV plot shows it is a tough call in the comparison of the procedures. In contrast,
 319 the regular CV does not offer such insight.

320 4.3. A summary measure based on PEC-CV

321 As we have shown, the PEC-CV contains much more info about the behaviors of the regression procedures in
 322 consideration than the regular 10-fold CV. The visual and intuitive representation assists decision making handily.
 323 Since it involves multiple DSRs at which the relative performances of the candidates can be quite different, sometimes
 324 we may want to have an overall measure that summarizes the winning frequencies of the candidates. Needless to
 325 say, there are various ways to do this, e.g., by considering different averagings (for instance, geometric mean versus
 326 algorithmic mean), possibly with a weight. Some theoretical results will be given in the next section.

327 In this subsection, we illustrate the use of one simple measure as an example to demonstrate the finite sample
 328 performance of the proposed approach. For various specific scenarios, other tailored measures may be more effective.
 329 Here, for each competing procedure, at each DSR, if its winning frequency is below 50%, it is replaced by zero, and
 330 then a simple average of the modified winning frequencies is taken. The candidate with a higher average is declared the
 331 winner. The modification helps to separate close competitors when we need to choose one. Again, the full PEC-CV
 332 plot provides info on strength of this resulted decision. Here, we consider three different model settings. Note that the
 333 first setting is chosen to be on classification and the square loss for regression is replaced by the 0-1 loss (classification
 334 error). In fact, the PEC-CV approach works generally for the generalized linear modeling frameworks where prediction
 335 accuracy of the response can be assessed based on the predictive negative log-likelihood or a sensible loss function
 336 such as 0-1 loss for classification.

337 Setting 1

338 At the sample size $n = 100, 200, 500$, for $0.6n$ samples with $Y = 0$, the covariate vector (X_1, X_2, X_3) follows $N(\mathbf{0}, \Sigma)$,
 339 where Σ is a 3×3 identity matrix; for the remaining $0.4n$ samples with $Y = 1$, we generate X_1, X_2, X_3 independently
 340 with $N(0.4, 1), N(0.3, 1), N(0, 1)$, respectively. We use CV methods to compare the Fisher's linear discriminant anal-
 341 ysis (LDA) [13] based on X_1 and X_2 with LDA based on all of the three predictors. To tell which procedure is truly
 342 better in classification, an independent testing data of size 50000 is used. Here naturally we consider the classification
 343 error (0-1 loss) instead of the square prediction error.

345 Setting 2

346 The model used to generate data has the nonlinear expression:

$$347 Y = X_1^2 + X_2^3 + X_3 + \varepsilon. \quad (1)$$

348 At the sample size $n = 100, 200, 500$, we draw the covariates (X_1, X_2, X_3) from $N(\mathbf{0}, \Sigma)$ with $\Sigma = (0.4^{|i-j|})_{i,j=1}^3$. For
 349 ε , we consider both $N(0, 1)$ and $t(3)$ distributions. We apply CV methods to compare RF with SVM based on the data
 350 generated by (1). Also, to tell which procedure is truly better in prediction, an independent testing data of size 50000
 351 is used.

352 Setting 3

353 We consider the following model for data generation:

$$354 Y = \beta_0 + \beta_1 X_1 + \beta_2 X_2 + \cdots + \beta_7 X_7 + \varepsilon.$$

355 In order to have realistic covariates, at the sample size $n = 100, 200, 500$, we randomly draw the covariates (X_1, \dots, X_8)
 356 from an air quality data set (<https://archive.ics.uci.edu/ml/datasets/Air+Quality>). The dataset contains
 357 9358 instances of hourly averaged responses from an array of 5 metal oxide chemical sensors embedded in an Air
 358 Quality Chemical Multisensor Device in an Italian city. We consider attributes PT08.S1-PT08.S5, Temperature, Rel-
 359 ative Humidity and Absolute Humidity as the independent variables. For the random error ε , independent of \mathbf{X} , we
 360 consider both $N(0, 1)$ and $t(3)$ distributions. We take $(\beta_0, \beta_1, \dots, \beta_7) = (0.5, 1, -1, 0.5, -0.5, 0.25, -0.25, 0.1)$ as an
 361 example where there are effects of different sizes. Note that X_8 is not needed for predicting Y . An independent testing
 362 data of size 5000 is extracted from the air quality data set to determine the truly better model. We apply CV methods
 363 to select between the linear regression model based on X_1, \dots, X_7 and that based on all of the eight predictors.

Setting 4

The data in this example are generated by

$$Y = \beta_0 + \beta_1 X_1 + \beta_2 X_2 + \cdots + \beta_{10} X_{10} + \varepsilon,$$

where $(\beta_0, \beta_1, \dots, \beta_{10}) = (0.5, 0.2, -0.5, 0.5, -1, 1, -1.5, 2, 0.5, -0.5, 1)$. The covariate vector (X_1, \dots, X_{16}) follows $N(0, \Sigma)$ with $\Sigma = (0.2^{|i-j|})_{i,j=1}^{16}$, and error term ε follows $N(0, 1)$. The sample size of this setting is 100, 200, 500.

Here we investigate the capabilities of the different CVs when the performance difference between the candidate methods varies. Besides the true mode (which correctly uses the first 10 variables), the model to compare (MTC) with is one of the following: 1) $Y \sim X_1 + X_2 + X_3 + \cdots + X_{16}$ (severely overfitting), 2) $Y \sim X_1 + X_2 + \cdots + X_{14}$ (overfitting), 3) $Y \sim X_2 + \cdots + X_{12}$ (less overfitting), 4) $Y \sim X_4 + \cdots + X_{11}$ (marginally overfitting). Clearly, the different choices of MTC here yield different degrees of competitiveness with respect to the true model. To verify that the true model is indeed better in prediction, an independent testing data of size 10000 is used.

We compare three methods: regular 10-fold CV, 10-fold EC-CV(24) and PEC-CV with the summary measure stated earlier. Here for EC-CV(r) involved in the calculation of PEC-CV, as in Example 4, we take $r = 240/k$ for k -fold with 4 choices: 5⁻¹-fold, 2-fold, 4-fold, and 10-fold. Both conditional and unconditional probabilities of choosing the better model are considered. For the conditional selection probability, each time we randomly draw the specified number of observations (n) of the covariates and response Y , and one of the two competing models is deemed to perform better based on the squared prediction error or classification error on the independent test data. If a CV method chooses this better model, it is regarded to have made the correct decision. Note that in Setting 3, although the smaller model is used to generate the data, it actually performs worse than the larger model sometimes. We replicate the data generation and selection process $N = 1000$ times and record the number of times each CV method chooses the conditionally better model. For unconditional selection, as explained earlier, we define the better model to be the one that minimizes the expected squared prediction error or the expected classification error on the test cases, which in our settings are the classifier based on X_1 and X_2 , RF and the true model, respectively. Clearly selecting the conditionally better model is more difficult. The selection results of Setting 1 to Setting 3 for unconditional and conditional better models are presented in Table 6 and Table 7. Similar results of Setting 4 are presented in Table 8 and Table 9. For Setting 4, in addition, we present the root mean square error (RMSE) of the candidate models (their standard errors are in the range 0.001 to 0.009) and their proportions of performing better in Table 10.

Note that the mean 0-1 loss of LDA based on the true model is 0.380 and that of LDA based on the larger model is 0.386. The difference may seem small, but a paired t -test shows that the performances are statistically significant at level 0.05. The results for Setting 4 show that in a range of competitiveness of the candidate models, the CV methods perform differently, from perfectly identifying the better model to selecting the worse model up to 34% in the worst case.

From these tables, we conclude that EC-CV improves over regular CV, but the performance of PEC-CV with the chosen averaging scheme is the best in terms of the proportion of identifying the better procedure (conditionally or unconditionally). Furthermore, our proposed methods can be used for classification as well, and are effective not only for the linear regression models, but also for the non-linear regression models with machine learning procedures as candidates.

5. Theoretical properties

In 1993, Shao [24] showed that surprisingly when comparing linear regression models, in order to identify the best model with high probability, the evaluation data size in CV must be dominating. When comparing general regression procedures, in 2007, Yang [31] showed that may not be necessary and the training data size can sometimes be dominating, depending on how competitive the candidates are (see [35] for further extensions). In this section, we mainly focus on cases where at least one of the candidate procedures is nonparametric with convergence rate (under squared error) slower than the parametric rate $1/n$. When only parametric models are considered, besides CV, we advocate the use of information criteria, proper testings, goodness of fit assessments, and model diagnostics for reaching a more reproducible and reliable conclusion (see, e.g., [21]).

Consider the regression setting:

$$Y_i = f(X_i) + \varepsilon_i, \quad 1 \leq i \leq n, \quad (2)$$

Table 6

Settings 1-3: Simulation results for unconditional winning proportion over $N = 1000$ replications. * indicates that PEC-CV performs significantly better than the other method at level 0.05.

Setting	Distribution	Method	$n = 100$	$n = 200$	$n = 500$
Setting 1	-	PEC-CV	0.818	0.778	0.782
		10-fold EC-CV	0.686*	0.754	0.658*
		10-fold CV	0.612*	0.574*	0.498*
Setting 2	$N(0, 1)$	PEC-CV	0.886	0.900	0.873
		10-fold EC-CV	0.802*	0.807*	0.772*
		10-fold CV	0.770*	0.780*	0.764*
	$t(3)$	PEC-CV	0.810	0.871	0.841
		10-fold EC-CV	0.724*	0.772*	0.744*
		10-fold CV	0.698*	0.760*	0.740*
Setting 3	$N(0, 1)$	PEC-CV	0.933	0.872	0.874
		10-fold EC-CV	0.865*	0.854	0.873
		10-fold CV	0.823*	0.806*	0.818*
	$t(3)$	PEC-CV	0.954	0.874	0.861
		10-fold EC-CV	0.888*	0.852	0.868
		10-fold CV	0.822*	0.816*	0.820*

Table 7

Settings 1-3: Simulation results for conditional winning proportion over $N = 1000$ replications. * indicates that PEC-CV performs significantly better than the other method at level 0.05.

Setting	Distribution	Method	$n = 100$	$n = 200$	$n = 500$
Setting 1	-	PEC-CV	0.662	0.606	0.586
		10-fold EC-CV	0.554*	0.534*	0.524
		10-fold CV	0.490*	0.510*	0.438*
Setting 2	$N(0, 1)$	PEC-CV	0.829	0.851	0.821
		10-fold EC-CV	0.750*	0.754*	0.724*
		10-fold CV	0.721*	0.726*	0.717*
	$t(3)$	PEC-CV	0.774	0.754	0.802
		10-fold EC-CV	0.680*	0.670*	0.704*
		10-fold CV	0.659*	0.640*	0.697*
Setting 3	$N(0, 1)$	PEC-CV	0.683	0.629	0.637
		10-fold EC-CV	0.621*	0.621	0.636
		10-fold CV	0.585*	0.580*	0.594*
	$t(3)$	PEC-CV	0.711	0.651	0.610
		10-fold EC-CV	0.656*	0.632	0.610
		10-fold CV	0.609*	0.611*	0.577*

410 where $(X_i, Y_i)_{i=1}^n$ are independent observations with X_i iid taking values in a d -dimensional Borel set $\mathcal{X} \subset R^d$ for
 411 some $d \geq 1$, f is the true regression function and ε_i is the error term that satisfies $E(\varepsilon_i|X_i) = 0$ and $E(\varepsilon_i^2)$ is finite.

Let δ_1 and δ_2 be two regression procedures that are to be compared. Let $\hat{f}_{n,1}(x)$ and $\hat{f}_{n,2}(x)$ denote the estimated regression function by the two procedures, respectively, at sample size n . For a chosen $1 < n_2 < n$ and $n_1 = n - n_2$, we split the data $(X_i, Y_i)_{i=1}^n$ into two parts $Z^1 = (X_i, Y_i)_{i=1}^{n_1}$, $Z^2 = (X_i, Y_i)_{i=n_1+1}^n$. Then δ_1 and δ_2 are trained on Z^1 to obtain $\hat{f}_{n_1,1}$ and $\hat{f}_{n_1,2}$ and we record the prediction errors

$$CV(\hat{f}_{n_1,j}) = \sum_{i=n_1+1}^n \left(Y_i - \hat{f}_{n_1,j}(X_i) \right)^2, \quad j = 1, 2. \quad (3)$$

Table 8

Setting 4: Simulation results for unconditional winning proportion over $N = 1000$ replications. * indicates that PEC-CV performs significantly better than the other method at level 0.05.

Setting	Distribution	Method	$n = 100$	$n = 200$	$n = 500$
MTC 1	$N(0, 1)$	PEC-CV	1.000	1.000	1.000
		10-fold EC-CV	1.000	1.000	1.000
		10-fold CV	1.000	1.000	1.000
	$t(3)$	PEC-CV	1.000	1.000	1.000
		10-fold EC-CV	1.000	1.000	1.000
		10-fold CV	1.000	1.000	1.000
MTC 2	$N(0, 1)$	PEC-CV	0.970	0.963	0.955
		10-fold EC-CV	0.905*	0.903*	0.916*
		10-fold CV	0.887*	0.885*	0.888*
	$t(3)$	PEC-CV	0.976	0.972	0.963
		10-fold EC-CV	0.928*	0.903*	0.921*
		10-fold CV	0.908*	0.901*	0.899*
MTC 3	$N(0, 1)$	PEC-CV	0.924	0.919	0.893
		10-fold EC-CV	0.851*	0.866*	0.852*
		10-fold CV	0.844*	0.858*	0.841*
	$t(3)$	PEC-CV	0.928	0.911	0.899
		10-fold EC-CV	0.851*	0.866*	0.865*
		10-fold CV	0.852*	0.836*	0.854*
MTC 4	$N(0, 1)$	PEC-CV	0.865	0.859	0.864
		10-fold EC-CV	0.829*	0.830*	0.845
		10-fold CV	0.814*	0.814*	0.841*
	$t(3)$	PEC-CV	0.873	0.858	0.854
		10-fold EC-CV	0.827*	0.830*	0.830*
		10-fold CV	0.817*	0.816*	0.815*

412 Now, given an integer $k \geq 2$, we split the data into equal-size (as much as possible) k parts. Then each part is taken
 413 as Z^2 in turn, while the rest as Z^1 to obtain the above prediction errors for δ_1 and δ_2 .

We sum up the total prediction errors over the k folds for each of δ_1 and δ_2 , and denote the total prediction errors as

$$TPE_k(j), \quad j = 1, 2. \quad (4)$$

We emphasize that here in obtaining TPE we tally up the prediction errors of different folds. Then let

$$W_k = \begin{cases} 1 & \text{if } TPE_k(1) \leq TPE_k(2), \\ 0 & \text{otherwise.} \end{cases} \quad (5)$$

414 That is, with the prediction errors tallied, δ_1 gets one vote if it has performed better for this specific k -fold data splitting.

Now let Π be a collection of permutations of the data, and let $W_k(\pi)$ be the voting result based on the permuted data with $\pi \in \Pi$. Then we count the total votes for δ_1 and calculate its ratio of winning (ROW):

$$ROW_k = \frac{1}{|\Pi|} \sum_{\pi \in \Pi} W_k(\pi). \quad (6)$$

415 If ROW_k is close to 1, it shows that δ_1 performs consistently better (over the data permutations) than δ_2 at the DSR
 416 $(k - 1) : 1$ (training:evaluation).

417 Now let us consider the reverse k -fold CV, denoted as k^{-1} -fold. The only difference from the above is that each
 418 time we use one fold for training and the remaining $k - 1$ folds for evaluation. Then the DSR is $1 : (k - 1)$ (train-
 419 ing:evaluation). The reason to consider a much smaller portion for training is that it in fact can be advantageous when

Table 9

Setting 4: Simulation results for conditional winning proportion over $N = 1000$ replications. * indicates that PEC-CV performs significantly better than the other at level 0.05.

Setting	Distribution	Method	$n = 100$	$n = 200$	$n = 500$
MTC 1	$N(0, 1)$	PEC-CV	1.000	1.000	1.000
		10-fold EC-CV	1.000	1.000	1.000
		10-fold CV	1.000	1.000	1.000
	$t(3)$	PEC-CV	1.000	1.000	1.000
		10-fold EC-CV	1.000	1.000	1.000
		10-fold CV	1.000	1.000	1.000
MTC 2	$N(0, 1)$	PEC-CV	0.949	0.957	0.951
		10-fold EC-CV	0.884*	0.897*	0.912*
		10-fold CV	0.866*	0.879*	0.884*
	$t(3)$	PEC-CV	0.961	0.969	0.962
		10-fold EC-CV	0.904*	0.925*	0.920*
		10-fold CV	0.893*	0.898*	0.898*
MTC 3	$N(0, 1)$	PEC-CV	0.837	0.865	0.873
		10-fold EC-CV	0.764*	0.812*	0.832*
		10-fold CV	0.757*	0.804*	0.821
	$t(3)$	PEC-CV	0.832	0.863	0.873
		10-fold EC-CV	0.779*	0.803*	0.839*
		10-fold CV	0.758*	0.788*	0.828*
MTC 4	$N(0, 1)$	PEC-CV	0.699	0.730	0.769
		10-fold EC-CV	0.667*	0.701*	0.750
		10-fold CV	0.658*	0.687*	0.0.746*
	$t(3)$	PEC-CV	0.711	0.718	0.783
		10-fold EC-CV	0.687*	0.687*	0.759*
		10-fold CV	0.661*	0.678*	0.744*

420 comparing close competitors (see, [24, 31, 35]). In this reverse k -fold case, the ratio of winning for δ_1 is denoted
 421 ROW_{k-1} .

422 The above is the newly proposed method of electoral college CV (note again that the electoral college system
 423 shares the spirit of totaling up over precincts in a state but then voting at the state level). As we have pointed out, it
 424 is beneficial to consider multiple DSRs to have a more comprehensive understanding of the comparison of the two
 425 procedures. Let \mathcal{K}_1 and \mathcal{K}_2 be two finite sets of positive integers. Let $\mathcal{K} = \{k, k \in \mathcal{K}_1\} \cup \{k^{-1} : k \in \mathcal{K}_2\}$. Then for
 426 $s \in \mathcal{K}$, if $s = k \in \mathcal{K}_1$, we obtain ROW_k ; if $1/s = k \in \mathcal{K}_2$, we obtain $ROW_{k-1} = ROW_s$. The ROW profile over
 427 $s \in \mathcal{K}$ provides much information on the two competing procedures. Direct graphs of the profile can visually offer an
 428 intuitive understanding, as we have shown. We may also numerically summarize the ROW values, as done below.

We define the average repeated ratio of winning ($ARROW$):

$$ARROW = \frac{1}{|\mathcal{K}|} \sum_{s \in \mathcal{K}} ROW_s. \tag{7}$$

Another way to summarize the comparison results at the different splitting ratios goes as follows: let

$$ARROW' = \frac{1}{|\mathcal{K}|} \sum_{s \in \mathcal{K}, ROW_s \geq 0.5} ROW_s. \tag{8}$$

429 Note that for an $s \in \mathcal{K}$ with $ROW_s < 0.5$, the winning frequency of δ_1 at this DSR is simply ignored in $ARROW'$.
 430 This version emphasizes more the degree of winning. For instance, if δ_1 is only slightly worse than δ_2 such that at
 431 each DSR, ROW_s for δ_1 is about 45%. Then the earlier $ARROW$ value is about 45%, correctly indicating that δ_1 is
 432 worse, but only slightly so. The $ARROW'$ statistic, however, has value 0, properly conveying the information that for
 433 the selection procedure, we are quite confident to choose δ_2 as the better one, regardless of how close behind δ_1 may
 434 be. So both versions of $ARROW$ provide useful info.

Table 10

Setting 4: Simulation results of RMSE of the candidate models and their proportions of performing better over $N = 1000$ replications.

		True	MTC 1	True	MTC 2	True	MTC 3	True	MTC 4	
$N(0, 1)$	$n = 100$	RMSE	0.345	0.469	0.345	0.416	0.345	0.363	0.345	0.354
		Better proportion	1.000	0.000	0.979	0.021	0.832	0.168	0.734	0.266
	$n = 200$	RMSE	0.237	0.295	0.237	0.281	0.237	0.248	0.237	0.240
		Better proportion	1.000	0.000	0.994	0.004	0.871	0.129	0.756	0.244
	$n = 500$	RMSE	0.147	0.184	0.147	0.174	0.147	0.154	0.147	0.254
		Better proportion	1.000	0.000	0.996	0.004	0.905	0.095	0.807	0.193
$t(3)$	$n = 100$	RMSE	0.570	0.794	0.570	0.683	0.570	0.600	0.570	0.581
		Better proportion	1.000	0.000	0.985	0.015	0.836	0.164	0.810	0.190
	$n = 200$	RMSE	0.402	0.484	0.402	0.479	0.402	0.422	0.402	0.415
		Better proportion	1.000	0.000	0.997	0.003	0.860	0.140	0.829	0.171
	$n = 500$	RMSE	0.247	0.308	0.247	0.263	0.247	0.259	0.247	0.251
		Better proportion	1.000	0.000	0.999	0.001	0.929	0.071	0.901	0.099

More generally, we may allow a weighting of the *ROW* values (e.g., large k values receive higher weights due to their being closer to the full sample size). Let $\mathbf{w} = (w_s : s \in \mathcal{K})$ be a weighting vector (i.e., $w_s \geq 0$ and $\sum_{s \in \mathcal{K}} w_s = 1$). Then define

$$ARROW(\mathbf{w}) = \sum_{s \in \mathcal{K}} w_s ROW_s. \tag{9}$$

While there are different ways to summarize the *ROW* values, we focus below on *ARROW* for our main theoretical result, which follows [31] for technical derivation.

Define the L_q norm

$$\|f\|_q = \begin{cases} \left(\int |f(x)|^q P_X(dx) \right)^{1/q} & \text{for } 1 \leq q < \infty \\ \text{esssup}|f| & \text{for } q = \infty, \end{cases}$$

where P_X denotes the probability distribution of X_1 . For the two competing procedures, we assume $\hat{f}_{n,1}$ and $\hat{f}_{n,2}$ converge exactly at rate p_n and q_n under the $\|\cdot\|_2$ loss, respectively, with $\max(p_n, q_n)$ converging more slowly than the parametric rate $1/\sqrt{n}$. Without loss of generality, we assume δ_1 is asymptotically better than δ_2 under the L_2 loss: $p_n = O(q_n)$ and for $\forall 0 < \varepsilon < 1$, \exists a constant c_ε , such that when n is large enough

$$P(\|\hat{f}_{n,2} - f\|^2 \geq (1 + c_\varepsilon)\|\hat{f}_{n,1} - f\|^2) \geq 1 - \varepsilon.$$

The following conditions are needed.

Condition 1. The error variances $E(\varepsilon_i^2 | X_i)$ are upper bounded by a constant $\bar{\sigma}^2 > 0$ almost surely for all $i \geq 1$.

Condition 2. For $j = 1, 2$,

$$\|f - \hat{f}_{n,j}\|_\infty = O_p(1). \tag{10}$$

Condition 3:

$$\frac{\|f - \hat{f}_{n,j}\|_4}{\|f - \hat{f}_{n,j}\|_2} = O_p(1), \quad j = 1, 2. \tag{11}$$

Theorem 1. Under the previous conditions, for every choice of \mathcal{K}_1 and \mathcal{K}_2 , for any number of data splittings and any weighting vector \mathbf{w} , we have

$$ARROW(\mathbf{w}) \rightarrow 1 \quad \text{in probability as } n \rightarrow \infty. \tag{12}$$

The same convergence holds for *ARROW'* and *ARROW*(\mathbf{w}).

Table 11
Specifications of datasets

Datasets	Type	p	n	n_{test}	RELM	SVM	
					C	C	γ
Housing	Reg	14	337	169	2^8	2^{-2}	2^0
Energy-cooling	Reg	8	512	256	2^{10}	2^8	2^0
Energy-heating	Reg	8	512	256	2^{21}	2^9	2^0
Mg	Reg	6	923	462	2^{10}	2^2	2^{-1}
Abalone	Reg	9	2784	1393	2^{15}	2^0	2^{-1}
Liver	Class	7	230	115	2^2	2^2	2^0
Breast cancer	Class	11	455	228	2^1	2^{-1}	2^{-1}
Australian	Class	15	460	230	2^2	2^2	2^{-1}
German	Class	25	666	334	2^4	2^0	2^0
Bank note	Class	5	914	458	2^1	2^2	2^0
Cpusmall	Reg	12	500	5000	2^{12}	2^5	2^0
Svmuide1	Class	5	500	2000	2^4	2^5	2^6

440 Thus, under the mild conditions stated, the *ARROW* measures will be on target with probability going to 1, i.e.,
 441 they will be close to 1 if δ_1 is better (and they will be close to 0 if δ_2 is better). Therefore, with a sufficiently large sample
 442 size, we should expect the PEC-CV to clearly show out the better procedure. Of course, in reality, the sample size may
 443 not be enough to distinguish two competitors decisively, and our *ARROW* statistics provide sensible quantifications
 444 on the relative performances of δ_1 and δ_2 .

445 6. Empirical study

446 To thoroughly evaluate the performances of the PEC-CV and EC-CV, benchmark datasets from the UCI machine
 447 learning Repository [5] and LIBSVM [10] are selected. The PEC-CV is compared to other kinds of CV (10-fold EC-
 448 CV(24), 10-fold CV, LOO). Here the CV methods are used to compare the performances of RELM [25] and SVM.
 449 To be consistent with previous ELM papers, the input weights are randomly generated from the range $[-1, 1]$ and the
 450 biases from $[0, 1]$. We employ Gaussian Radial Basis Functions (RBF) as the activation function. For large data sets
 451 (Cpusmall and Svmguide1), we split the data into two parts, a final testing sample size of n_{test} and the rest for selecting
 452 the training samples. At the sample size of $n = 500$, each time, we randomly extract n observations from the training
 453 part. For small data sets, we directly split the data into two parts, a final testing sample of size n_{test} and a data set of
 454 size n , and a random data permutation is performed before each splitting of data. A total of $N = 100$ trials are carried
 455 out for each data set. The training and testing data sets are divided as indicated in Table 11. The tuning parameters
 456 of the RELM are the same as in [25], which are also reported in Table 11. Note that winning proportion is called for
 457 short.

458 Table 12 presents the comparisons of the regular 10-fold CV, 10-fold EC-CV(24), and a version of PEC-CV as
 459 explained in Section 4, under conditional and unconditional selections respectively. Here for EC-CV(r) involved in
 460 the calculation of PEC-CV, as in Example 4, we take $r = 240/k$ for k -fold with 4 choices: 5^{-1} -fold, 2-fold, 4-fold, and
 461 10-fold. The results show that 10-fold EC-CV improves over the regular 10-fold CV, but PEC-CV further improves
 462 the performances.

463 To conclude, the real data example has illustrated that the PEC-CV can lead to significantly improved performance
 464 over the regular CV. Another real example with similar findings can be found in the supplementary file.

Table 12

Real data analysis results using PEC-CV, 10-fold EC-CV, 10-fold CV and LOO over $N = 100$ replications.

Datasets	Type	Mean RMSE		PEC-CV		10-fold EC-CV		10-fold CV		LOO	
		RELM	SVM	WP		WP		WP		WP	
				cond	uncond	cond	uncond	cond	uncond	cond	uncond
Housing	Reg	0.076	0.087	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000
Energy-cooling	Reg	0.071	0.079	1.000	1.000	1.000	1.000	0.950	0.950	0.800	0.800
Energy-heating	Reg	0.074	0.088	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000
Mg	Reg	0.134	0.135	0.710	0.750	0.640	0.660	0.610	0.650	0.570	0.610
Abalone	Reg	0.078	0.076	0.650	0.710	0.650	0.710	0.520	0.550	0.520	0.550
Cpusmall	Reg	0.055	0.059	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000
		Mean Accuracy		WP		WP		WP		WP	
		RELM	SVM	cond		uncond		cond		uncond	
				cond	uncond	cond	uncond	cond	uncond	cond	uncond
Liver	Class	0.713	0.643	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000
Breast cancer	Class	0.970	0.960	0.830	0.930	0.820	0.910	0.65	0.710	0.800	0.900
Australian	Class	0.857	0.852	0.760	1.000	0.680	0.910	0.590	0.840	0.570	0.820
German	Class	0.734	0.725	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000
Bank note	Class	1.000	0.987	0.860	1.000	0.860	1.000	0.730	0.930	0.730	0.930
Svmguide1	Class	0.966	0.954	0.750	0.800	0.750	0.800	0.650	0.700	0.710	0.730

7. Conclusion and discussion

CV remains the most widely used tool to choose a supervised learning procedure. Here the goal is to single out the candidate that has the best prediction accuracy for future applications. The traditional k -fold CV, the popular 10-fold in particular, has three major drawbacks in terms of stability, a lack of a reliability measure/index, and inability to reflect the dynamic relative performances of the competing learning procedures and the assessments as the sample size changes. The proposed PEC-CV addresses these difficulties by repeated data splittings at multiple data splitting ratios and an integration of averaging (of prediction errors over different folds for each k -fold data partition) and voting. The averaging part enhances the prediction accuracy measure and the voting part provides the previously unavailable valuable information on competitiveness of the candidates. Under sensible conditions, the winning frequencies of the best candidate should stand out, approaching 1 theoretically as the sample size increases. The profile of the winning frequencies yields much insight on the choice of the best candidate as we have illustrated in the paper. In contrast, a pure averaging (e.g., repeated k -fold) suffers from a lack of reliable quantification of how much better the winning procedure is over the others.

As we mentioned in the introduction, our proposed CV method is computationally more demanding due to the use of multiple DSRs. It is nice that there are some modified CV methods like the approximated cross-validation based on Bouligand influence function (BIF) in [19] being computed efficiently for certain kernel-based methods, and we can apply our PEC-CV strategy to this approximated BIF based criterion function by replacing $TPE_k(j)$, $j = 1, 2$ in this paper with $BIF_k^r(j)$, $j = 1, 2$ in [19]. In this way, we can save time at each DSR, thereby reducing the total running time for using CV with kernel-based algorithms.

It should be emphasized that our focus in the paper is on choosing the best candidate. The conclusions do not necessarily apply to prediction error estimation itself (see, e.g., [6, 20, 22, 35]).

A. Appendix: Proof of Theorem 1

PROOF OF THEOREM 1. The main idea and technical derivation follow from [31], which handles a purely voting-based CV at a single DSR. We give a sketched proof and more details can be found in [31].

Under the conditions given, from [31], we know that for each fixed DSR n_2/n_1 , as is true for the k -fold and k^{-1} -fold versions,

$$P\left(CV(\hat{f}_{n_1,1}) \leq CV(\hat{f}_{n_1,2})\right) \rightarrow 1 \text{ in probability.}$$

491 Let $CV(\hat{f}_{n_1,j}^{(l)})$ denote the sum of prediction errors when the l -th fold is used as the evaluation part, $j = 1, 2$. Then

$$\begin{aligned} & P(TPE_k(1) > TPE_k(2)) \\ &= P\left(\sum_{l=1}^k CV(\hat{f}_{n_1,1}^{(l)}) > \sum_{l=1}^k CV(\hat{f}_{n_1,2}^{(l)})\right) \\ &\leq P\left(\bigcup_{l=1}^k \{CV(\hat{f}_{n_1,1}^{(l)}) > CV(\hat{f}_{n_1,2}^{(l)})\}\right) \\ &\leq kP\left(CV(\hat{f}_{n_1,1}) > CV(\hat{f}_{n_1,2})\right) \rightarrow 0 \text{ as } n \rightarrow \infty. \end{aligned}$$

492 Clearly, the same argument applies to the k^{-1} -fold case.

493 Thus, we have shown that for every k (and k^{-1}), $W_k \rightarrow 1$ in probability. Consequently, since $0 \leq W_k \leq 1$ almost
494 surely, we must have $EW_k \rightarrow 1$ as $n \rightarrow \infty$. Then

$$E\left(\frac{1}{|\Pi|} \sum_{\pi \in \Pi} W_k(\pi)\right) = EW_k \rightarrow 1.$$

495 Again, because $0 \leq \frac{1}{|\Pi|} \sum_{\pi \in \Pi} W_k(\pi) \leq 1$ almost surely, the above convergence implies

$$\frac{1}{|\Pi|} \sum_{\pi \in \Pi} W_k(\pi) \rightarrow 1 \text{ in probability,}$$

496 i.e., $ROW_k \rightarrow 1$ in probability.

497 Since this holds for each k -fold or k^{-1} -fold, we conclude $ARROW \rightarrow 1$ in probability, and the same conclusion
498 also holds for $ARROW'$ and $ARROW(\mathbf{w})$ for every weighting vector \mathbf{w} . This completes the proof of the theorem.

CRedit authorship contribution statement

Zishu Zhan: Methodology, numerical work, writing. **Yuhong Yang:** Conceptualization of this study, methodology, writing.

Declaration of Competing Interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Acknowledgments

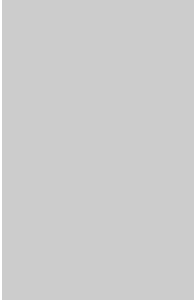
This research did not receive any specific grant from funding agencies in the public, commercial, or not-for-profit sectors. We sincerely thank the Editor and the reviewers for carefully reading our paper and providing truly insightful and valuable suggestions to improve our work in both content and presentation.

References

- [1] Allen, D.M., 1974. The relationship between variable selection and data augmentation and a method for prediction. *Technometrics* 16, 125–127. doi:10.1080/00401706.1974.10489157.
- [2] Alpaydin, E., 1999. Combined 5×2 cv f test for comparing supervised classification learning algorithms. *Neural Computation* 11, 1885–1892. doi:10.1162/089976699300016007.
- [3] Arlot, S., Celisse, A., 2009. A survey of cross-validation procedures for model selection. *Statistics Surveys* 4, 40–79. doi:10.1214/09-SS054.
- [4] Arlot, S., Lerasle, M., 2016. Choice of v for v -fold cross-validation in least-squares density estimation. *Journal of Machine Learning Research* 17, 1–50. URL: <http://www.jmlr.org/papers/volume17/14-296/14-296.pdf>.
- [5] Asuncion, A., Newman, D., 2007. Uci machine learning repository. URL: <https://archive.ics.uci.edu/ml/index.php>.
- [6] Bengio, Y., Grandvalet, Y., 2004. No unbiased estimator of the variance of k -fold cross-validation. *Journal of Machine Learning Research* 5, 1089–1105. URL: <http://www.jmlr.org/papers/volume5/grandvalet04a/grandvalet04a.pdf>.

- [7] Breiman, L., 2001. Random forests. *Machine Learning* 45, 5–32. doi:10.1023/A:1010933404324.
- [8] Breiman, L., Friedman, J.H., Olshen, R.A., Stone, C.J., 1984. *Classification and regression trees*. Wadsworth, Monterey.
- [9] Burman, P., 1989. A comparative study of ordinary cross-validation, v-fold cross-validation and the repeated learning-testing methods. *Biometrika* 76, 503–514. doi:10.1093/BIOMET/76.3.503.
- [10] Chang, C.C., Lin, C.J., 2011. Libsvm: a library for support vector machines. *ACM Transactions on Intelligent Systems and Technology* 2, 1–27. doi:10.1145/1961189.1961199.
- [11] Cortes, C., Vapnik, V., 1995. Support-vector networks. *Machine learning* 20, 273–297. doi:10.1023/A:1022627411411.
- [12] Ding, J., Tarokh, V., Yang, Y., 2018. Model selection techniques: An overview. *IEEE Signal Processing Magazine* 35, 16–34. doi:10.1109/MSP.2018.2867638.
- [13] Fisher, R.A., 1936. The use of multiple measurements in taxonomic problems. *Annals of Human Genetics* 7, 179–188. doi:10.1111/J.1469-1809.1936.TB02137.X.
- [14] Geisser, S., 1975. The predictive sample reuse method with applications. *Journal of the American Statistical Association* 70, 320–328. doi:10.1080/01621459.1975.10479865.
- [15] Hurvich, C.M., Tsai, C.L., 1989. Regression and time series model selection in small samples. *Biometrika* 76, 297–307. doi:10.1093/biomet/76.2.297.
- [16] Jiang, G., Wang, W., 2017. Error estimation based on variance analysis of k-fold cross-validation. *Pattern Recognition* 69, 94–106. doi:10.1016/J.PATCOG.2017.03.025.
- [17] Kohavi, R., 1995. A study of cross-validation and bootstrap for accuracy estimation and model selection, in: *In Proceedings of the Fourteenth International Joint Conference on Artificial Intelligence*, pp. 1137–1143. URL: <https://www.ijcai.org/Proceedings/95-2/Papers/016.pdf>.
- [18] Liu, W., Yang, Y., 2011. Parametric or nonparametric? a parametricness index for model selection. *The Annals of Statistics* 39, 2074–2102. doi:10.1214/11-AOS899.
- [19] Liu, Y., Liao, S., Jiang, S., Ding, L., Lin, H., Wang, W., 2020. Fast cross-validation for kernel-based algorithms. *IEEE Transactions on Pattern Analysis and Machine Intelligence* 42, 1083–1096. doi:10.1109/TPAMI.2019.2892371.
- [20] Markatou, M., Tian, H., Biswas, S., Hripcsak, G., 2005. Analysis of variance of cross-validation estimators of the generalization error. *Journal of Machine Learning Research* 6, 1127–1168. doi:10.7916/D86D5R2X.
- [21] Nan, Y., Yang, Y., 2014. Variable selection diagnostics measures for high-dimensional regression. *Journal of Computational and Graphical Statistics* 23, 636–656. doi:10.1080/10618600.2013.829780.
- [22] Rodriguez, J., Perez, A., Lozano, J., 2010. Sensitivity analysis of k-fold cross validation in prediction error estimation. *IEEE Transactions on Pattern Analysis and Machine Intelligence* 32, 569–575. doi:10.1109/TPAMI.2009.187.
- [23] Schwarz, G., 1978. Estimating the dimension of a model. *The Annals of Statistics* 6, 461–464. doi:10.1214/AOS/1176344136.
- [24] Shao, J., 1993. Linear model selection by cross-validation. *Journal of the American Statistical Association* 88, 486–494. doi:10.1080/01621459.1993.10476299.
- [25] Shao, Z., Er, M.J., 2016. Efficient leave-one-out cross-validation-based regularized extreme learning machine. *Neurocomputing* 194, 260–270. doi:10.1016/J.NEUCOM.2016.02.058.
- [26] Stone, M., 1974. Cross-validation choice and assessment of statistical predictions (with discussion). *Journal of the Royal Statistical Society: Series B (Methodological)* 36, 111–133. doi:10.1111/J.2517-6161.1974.TB00994.X.
- [27] Sugiyama, M., Krauledat, M., Müller, K., 2007. Covariate shift adaptation by importance weighted cross validation. *Journal of Machine Learning Research* 8, 985–1005. URL: <http://www.jmlr.org/papers/volume8/sugiyama07a/sugiyama07a.pdf>.
- [28] Tibshirani, R., 1996. Regression shrinkage and selection via the lasso. *Journal of the Royal Statistical Society: Series B (Methodological)* 58, 267–288. doi:10.1111/J.2517-6161.1996.TB02080.X.
- [29] Wang, J., 2010. Consistent selection of the number of clusters via cross-validation. *Biometrika* 97, 893–904. doi:10.1093/BIOMET/ASQ061.
- [30] Yang, Y., 2006. Comparing learning methods for classification. *Statistica Sinica* 16, 635–657. URL: <http://www3.stat.sinica.edu.tw/statistica/oldpdf/A16n216.pdf>.
- [31] Yang, Y., 2007. Consistency of cross-validation for comparing regression procedures. *The Annals of Statistics* 35, 2450–2473. doi:10.1214/009053607000000514.
- [32] Yu, B., Kumbier, K., 2019. Three principles of data science: predictability, computability, and stability (pcs) URL: <https://arxiv.org/abs/1901.08152v1>.
- [33] Yu, Y., Feng, Y., 2014. Modified cross-validation for penalized high-dimensional linear regression models. *Journal of Computational and Graphical Statistics* 23, 1009–1027. doi:10.1080/10618600.2013.849200.
- [34] Zhang, P., 1993. Model selection via multifold cross validation. *The Annals of Statistics* 21, 299–313. doi:10.1214/AOS/1176349027.
- [35] Zhang, Y., Yang, Y., 2015. Cross-validation for selecting a model selection procedure. *Journal of Econometrics* 187, 95–112. doi:10.1016/J.JECONOM.2015.02.006.

Profile Electoral College Cross-Validation



Zishu Zhan is a Ph.D. student in the School of Statistics in Renmin University. She received her M.S. degree from Renmin University in 2020. Her research interests include model averaging, missing data problems and machine learning.

Yuhong Yang received his Ph.D. degree in statistics from Yale University in 1996, and has been a full professor in School of Statistics at the University of Minnesota since 2007. His research interests include model selection, multi-armed bandit problems, information theory, high-dimensional data analysis, and machine learning. He has published in journals in several fields including *Annals of Statistics*, *IEEE Transactions on Information Theory*, *IEEE Signal Processing Magazine*, *Journal of Econometrics*, *Journal of Machine Learning Research*, and *International Journal of Forecasting*. He is a fellow of the Institute of Mathematical Statistics and a recipient of the NSF CAREER Award.