Full versus incomplete cross-validation: measuring the impact of imperfect separation between training and test sets in prediction error estimation

Roman Hornung

Joint work with Christoph Bernau, Caroline Truntzer, Thomas Stadler and Anne-Laure Boulesteix

LMU Munich
Department of Medical Informatics, Biometry and Epidemiology

July, 7th, 2014

1. Introduction
2. Prediction rules
3. Error frequency and (incomplete) CV
4. New measure CVIIM
5. Addon procedures
6. Illustration
7. Simulation results
8. Summary & Conclusion

Introduction

- Modern technologies, most prominently microarrays, enable the measurement of the expression of **thousands or many thousands of genes for each unit of investigation**.

- Agglomerating such measurements for units (patients, tissues,...) which are affected by a **disease of interest** and for unaffected controls enables the building of **prediction rules** for the purpose of predicting the status of new units.

- Due to limited sample sizes and information contained in gene expression such prediction rules make errors.

- Our general interest lies in the **estimation of the expected error frequency**.

Prediction rule in general

Sample: $\boldsymbol{S} = \{(\boldsymbol{X}_1, Y_1), \ldots, (\boldsymbol{X}_n, Y_n)\} \sim P^n$

$\boldsymbol{X}_i$: ($LONG$) vector of (raw!) gene expressions (“**covariates**”)

Y_i : status of patient (“**class**”) ($i = 1, \ldots, n$)

Prediction rule fitted on “**training data**” $\boldsymbol{S}$:

$$\hat{g}_{\boldsymbol{S}} : \mathcal{X} \mapsto \mathcal{Y} = \{1, 2\}$$

Predicted status of new patient with gene expression vector $\boldsymbol{x} \in \mathcal{X}$:

$$\hat{g}_{\boldsymbol{S}}(\boldsymbol{x}) = \hat{y}$$

Exemplary analysis for obtaining a prediction rule

- Data material: 100 DNA microarrays of breast tissues, 50 affected with early stage breast cancer and 50 unaffected

- Analysis:
  1. Normalization using the RMA method. ⇒ 47,000 different expression variables
  2. t-test based selection of the 500 most informative variables
  3. Cross-validation based selection of the optimal cost parameter for the Support Vector Machine (SVM) classification method
  4. Fitting the SVM classification method using the optimized cost parameter

⇒ Three preliminary steps before fitting the actual classification method - these steps are part of prediction rule $\hat{g}_{\boldsymbol{s}}(\cdot)$.

(Expected) misclassification probability

- Misclassification probability of $\hat{g}_{\boldsymbol{S}}(\cdot)$:

$$\varepsilon[\hat{g}_{\boldsymbol{S}}] := \mathbb{P}_{(\boldsymbol{X},Y)\sim P}[\hat{g}_{\boldsymbol{S}}(\boldsymbol{X}) \neq Y]$$

Relevant to the medical doctor

- Expected misclassification probability when considering samples of size n (following distribution P^n):

$$\varepsilon(n) := \mathbb{E}_{\boldsymbol{S}\sim P^n}[\,\varepsilon[\hat{g}_{\boldsymbol{S}}]\,]$$

Relevant to the statistical methodologist

Motivation for cross-validation

- Taking the misclassification rate of $\hat{g}_{\boldsymbol{S}}(\cdot)$ on $\boldsymbol{S}$ as an estimator for $\varepsilon[\hat{g}_{\boldsymbol{S}}]$ would result in an unrealistically small error estimate, because $\hat{g}_{\boldsymbol{S}}(\cdot)$ is overly adapted to $\boldsymbol{S}$ ("resubstitution bias").
- Building a prediction rule on only a part $\boldsymbol{S}_{train} \subset \boldsymbol{S}$ of the data and estimating $\varepsilon[\hat{g}_{\boldsymbol{S}_{train}}]$ on the rest $\boldsymbol{S}_{test} = \boldsymbol{S}/\boldsymbol{S}_{train}$ often very inefficient because of limited sample size

$\Rightarrow$ Motivation for cross-validation, which is an estimator for $\varepsilon(n_{train}) := \mathbb{E}_{\boldsymbol{S}_{train} \sim P^{n_{train}}}[\,\varepsilon[\hat{g}_{\boldsymbol{S}_{train}}]\,]$ with $n_{train} < n$.

Cross-validation (CV) procedure

- General Idea: Perform more than one splitting of the whole data set $\boldsymbol{S}$ into two parts $\boldsymbol{S}_{train}$ and $\boldsymbol{S}_{test}$
- Procedure:
  1. Split the data set $\boldsymbol{S}$ into K (approximately) equally sized folds $\boldsymbol{S}_1, \dots, \boldsymbol{S}_K$
  2. For $k = 1, \dots, K$: Use the units in $\boldsymbol{S}/\boldsymbol{S}_k$ for constructing the prediction rule and the units in $\boldsymbol{S}_k$ as test data.
  3. Average the misclassification rates out of the K splittings in (2).
- Formula of the CV estimator $e_{full,K}(\boldsymbol{S})$:

$$\frac{1}{K}\sum_{k=1}^{K}\frac{1}{\#\boldsymbol{S}_k}\sum_{j \in \{i \,:\, (\boldsymbol{X}_i, Y_i) \in \boldsymbol{S}_k\}} I(\hat{g}_{\boldsymbol{S}\setminus\boldsymbol{S}_k}(\boldsymbol{X}_j) \neq Y_j),$$

In practice: repeat and take the average

Properties of cross-validation (CV)

- (Formally) not an estimator for the misclassification probability of a specific prediction rule (i.e. $\varepsilon[\hat{g}_{\boldsymbol{S}_{train}}]$), but one of the expected misclassification probability of samples of size $n_{train,K} := \#\{\boldsymbol{S}/\boldsymbol{S}_k\}$ ($k \in \{1, \ldots, K\}$) (i.e. $\varepsilon(n_{train,K})$)
- For $n_{train,K}$ approaching n (big K) its expectancy gets increasingly similar to $\varepsilon[\hat{g}_{\boldsymbol{S}}]$ - interesting for the medical doctor, but unfavourable for the methodologist
- High variance (around $\varepsilon(n_{train,K})$)

Incomplete cross-validation

- Incomplete CV: One or more preliminary steps performed before CV
  $\Rightarrow$ Part of the prediction rule already constructed on the whole data set
- Violation of the training and test set principle of CV
- Can lead to severe underestimation of the expected misclassification probability
  $\Rightarrow$ Over-optimistic conclusions possible
- Incomplete CV known to be severely downwardly biased for the case of variable selection

Incomplete cross-validation

- Issue previously unexamined for other preliminary steps in the literature (to our knowledge)
- Preliminary steps almost always conducted before CV

  Examples: normalization of gene expression data, imputation of missing values, variable filtering by variance, dichotomization of continuous variables, data-driven determination of powers of fractional polynomials, sophisticated preprocessing steps for imaging data, ...
- No certainty on the extent of downward bias through incomplete CV with respect to such steps

Incomplete cross-validation

- Formula of the incomplete CV estimator $e_{incompl,K}(\boldsymbol{S})$:

$$\frac{1}{K}\sum_{k=1}^{K}\frac{1}{\#\boldsymbol{S}_k}\sum_{j\,\in\,\{i\,:\,(\boldsymbol{X}_i,Y_i)\,\in\,\boldsymbol{S}_k\}} I(\hat{g}^{\boldsymbol{S}}_{\boldsymbol{S}\setminus\boldsymbol{S}_k}(\boldsymbol{X}_j)\neq Y_j),$$

where $\hat{g}^{\boldsymbol{S}}_{\boldsymbol{S}\setminus\boldsymbol{S}_k}(\cdot)$ denotes the prediction rule obtained when specific steps in its construction on $\boldsymbol{S}\setminus\boldsymbol{S}_k$ are performed on the whole sample $\boldsymbol{S}$.

Incomplete cross-validation

- $e_{incompl,K}(\boldsymbol{S})$ is a negatively biased estimator for $\varepsilon(n_{train,K})$, while CV $e_{full,K}(\boldsymbol{S})$ is unbiased for $\varepsilon(n_{train,K})$.
- $e_{incompl,K}(\boldsymbol{S})$ is unbiased as an estimator of the following term:

$$\varepsilon_{incompl}(n_{train,K}; n) :=$$

$$\mathbb{E}_{\boldsymbol{S}\sim P^n}\left\{ \mathbb{P}[\hat{g}^{\boldsymbol{S}}_{\boldsymbol{S}_{train,K}}(\boldsymbol{X}_{n_{train,K}+1}) \neq Y_{n_{train,K}+1}] \right\},$$

with $\boldsymbol{S}_{train,K} := \{(\boldsymbol{X}_1, Y_1), \ldots, (\boldsymbol{X}_{n_{train,K}}, Y_{n_{train,K}})\} \subset \boldsymbol{S}$ and $(\boldsymbol{X}_{n_{train,K}+1}, Y_{n_{train,K}+1}) \subset \boldsymbol{S}$ playing the role of an arbitrary test set observation.

Full versus incomplete cross-validation

CV = "Full CV"

Perform all steps for obtaining the prediction rule within CV.

$\Rightarrow e_{full,K}(\boldsymbol{S})$

Incomplete CV

Perform one or more steps before CV on the whole data set. ↯ formally wrong

$\Rightarrow e_{incompl,K}(\boldsymbol{S})$

In general likely: $e_{incompl,K}(\boldsymbol{S}) < \varepsilon(n_{train,K})$.

↯ Full CV can be computationally intensive and procedures to integrate certain steps into CV are often not implemented.

BUT: The extent to which $e_{incompl,K}(\boldsymbol{S})$ underestimates $\varepsilon(n_{train,K})$ can be marginal in some cases.

$\Rightarrow$ Full CV can be avoided in these cases.

A new measure of the impact of CV incompleteness (CVIIM)

- Quantitative measure for the degree of bias induced by incomplete CV with respect to specific steps.

- Main purpose: Spot cases, where full CV can be avoided generally and cases where incomplete CV is especially dangerous.

- Straightforward, but naive measure would be:
  $\varepsilon(n_{train,K}) - \varepsilon_{incompl}(n_{train,K}; n)$
  ↯ Smaller differences can easily also be due to a smaller $\varepsilon(n_{train,K})$.
  $\Rightarrow$ Preference for the ratio of the errors

A new measure of the impact of CV incompleteness (CVIIM)

Our new measure CVIIM (standing for "Cross-Validation Incompleteness Impact Measure") is defined as

$$\mathrm{CVIIM}_{P,n,K} = \begin{cases} 1 - \dfrac{\varepsilon_{incompl}(n_{train,K}; n)}{\varepsilon(n_{train,K})} & \text{if } \varepsilon_{incompl}(n_{train,K}; n) < \varepsilon(n_{train,K}) \text{ and } \varepsilon(n_{train,K}) > 0 \\ 0 & \text{otherwise} \end{cases}$$

$\in [0, 1]$. Larger values of $\mathrm{CVIIM}_{P,n,K}$ are associated with a stronger underestimation of $\varepsilon(n_{train,K})$.

Interpretation: Relative reduction of mean estimated error when performing incomplete CV

A new measure of the impact of CV incompleteness (CVIIM)

- Estimator of $\text{CVIIM}_{P,n,K}$: Replace $\varepsilon(n_{train,K})$ and $\varepsilon_{incompl}(n_{train,K}; n)$ by their unbiased estimators $e_{full,K}(\boldsymbol{S})$ and $e_{incompl,K}(\boldsymbol{S})$; denoted as $\text{CVIIM}_{\boldsymbol{S},n,K}$
- Rules of thumb: $\text{CVIIM}_{\boldsymbol{S},n,K} \in$ $[0, 0.02] \Rightarrow \sim$ no bias, $]0.02, 0.1] \Rightarrow$ weak bias, $]0.1, 0.2] \Rightarrow$ medium bias, $]0.2, 0.4] \Rightarrow$ strong bias, $]0.4, 1] \Rightarrow$ very strong bias.
- $\text{CVIIM}_{P,n,K}$ dependent on data distribution P $\Rightarrow$ Calculate $\text{CVIIM}_{\boldsymbol{S},n,K}$ for several data sets and average the values to draw conclusions.

“Addon procedures” for preliminary steps

- In general: for the purpose of prediction each step performed for obtaining a prediction rule has to be done on new units as well

- Naive approach: 1) Pool training data with new units, 2) re-perform all preliminary steps, 3) fit the classification method anew on the training data

  ↯ a) Impossible for steps the fitting of which requires the target variable; b) prediction rule is commonly kept fixed

  $\Rightarrow$ All steps have to be integrated into the constructed prediction rule $\hat{g}_{\boldsymbol{S}}(\cdot)$.

“Addon procedures” for preliminary steps

- “Addon procedures”: New units made subject to exactly the same procedure as those in the training data, but new units not involved in the adaption of the procedure to the training data.

- Example - Addon procedure for variable selection:

  The same variables are chosen for new units than for those in the training data, but only the training data is used to determine, which variables are used.

Illustration

- Various real-life data sets, mostly gene-expression data
- Investigated preliminary steps:

  1) variable selection, 2) variable filtering by variance, 3) choice of tuning parameters for various classification methods, 4) imputation using a variant of k-Nearest-Neighbors, 5) normalization with the RMA method
- In each case (incomplete) CVs repeated 300 times and the results averaged.
- Splitting ratios between the sizes of the training and test sets: 2:1 (3-fold CV), 4:1 (5-fold CV) and 9:1 (10-fold CV)

Overview of used data sets

Name	number of samples	number of variables	% diseased	type of variables	disease
ProstatecTranscr	102	12,625	51%	transcriptomic	prostate cancer
HeadNeckcTranscr	50	22,011	50%	transcriptomic	head and neck squamous
LungcTranscr	100	22,277	49%	transcriptomic	lung Adenocarcinoma
SLETranscr	36	47,231	56%	transcriptomic	systemic lupus erythematosus
GenitInfCoww0	51	21	71%	various	genital infection in cows
GenitInfCoww1	51	24	71%	various	genital infection in cows
GenitInfCoww2	51	27	71%	various	genital infection in cows
GenitInfCoww3	51	26	71%	various	genital infection in cows
GenitInfCoww4	51	27	71%	various	genital infection in cows
ProstatecMethyl	70	222	41%	methylation	prostate cancer
ColoncTranscr	47	22,283	53%	transcriptomic	colon cancer
WilmsTumorTranscr	100	22,283	42%	transcriptomic	Wilms' tumor

Investigated steps: variable selection with addon

- Procedure:
  1. For each variable: two-sample t-tests with groups according to the target variable
  2. Select the p_{sel} variables with the smallest p-values out of the t-tests
- Considered values for number of selected variables p_{sel}: 5, 10, 20 and half of the total number p of variables
- Classification methods: Diagonal Linear Discriminant Analysis (DLDA) for $p_{sel} = p/2$, otherwise Linear Discriminant Analysis (LDA)

Addon procedure:

- Use only the variables, which were selected on the training data

Investigated steps: variable filtering by variance with addon

- Procedure: Calculate the empirical variance of every variable and select the $p/2$ variables with the largest variances.
- Classification method: DLDA

Addon procedure:

- As with the t-test-based variable selection use only the variables selected on the training data.

Investigated steps: optimization of tuning parameters with addon

- Optimization of tuning parameters on a grid for seven different classification methods:

1. number of iterations m_{stop} in componentwise boosting with logistic loss function
2. number of neighbors in the k-Nearest-Neighbors algorithm
3. shrinkage intensity in Lasso
4. shrinkage intensity for the class centroids in Nearest Shrunken Centroids
5. number of components in Linear Discriminant Analysis on Partial Least Squares components
6. number *mtry* of variables randomly sampled as candidates at each split in Random Forests
7. cost parameter in Support Vector Machines with linear kernel

Investigated steps: optimization of tuning parameters with addon

- Procedure: For each candidate value of the tuning parameter on a respective grid, 3-fold CV (i.e. $K=3$) of the classifier is performed using this value of the tuning parameter. The value yielding the smallest 3-fold CV error is selected.

Addon procedure:

- The tuning parameter value chosen on the training data is used.

Investigated steps: Imputation of missing values with addon

- Procedure: *k*-Nearest-Neighbors imputation with standardization during the imputation; Tuning of *k* using 3-fold CV in an analogous way as described before

- Classification method: Nearest Shrunken Centroids for the high-dimensional data set (of those considered for this step) and Random Forests for the other data sets

Addon procedure:

- For the standardization use means and standard deviations estimated from the training data.

- Search *k* nearest neigbours only on the training data.

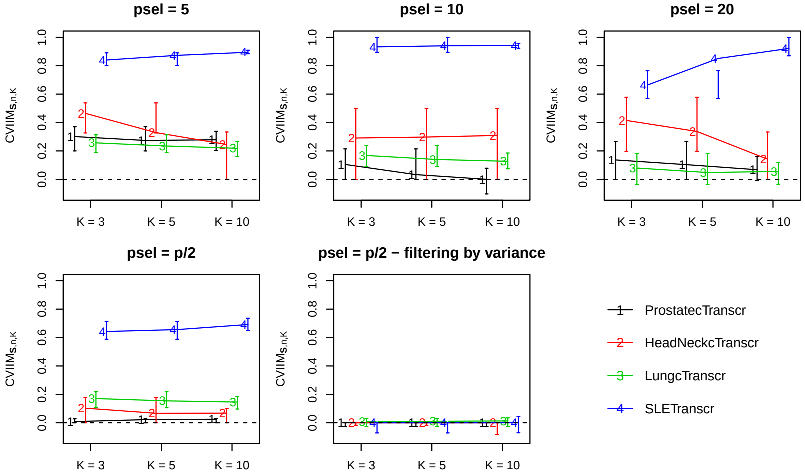
Investigated steps: Robust Multi-array Average (RMA) with addon

- Purpose: remove technical artefacts in raw microarray data and summarize the multiple measurements done for each variable value
- Three steps: 1) Background correction, 2) Quantile normalization, 3) Summarization
- Classification method: Nearest Shrunken Centroids

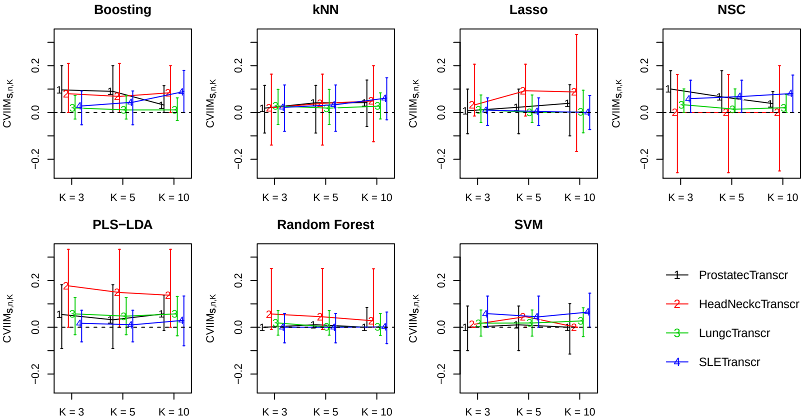
Addon procedure:

- Steps 1) and 3) performed array by array. ⇒ Only 2) requires an addon strategy.
- Use quantiles from training data to perform quantile normalization for new samples (Kostka and Spang, 2008).

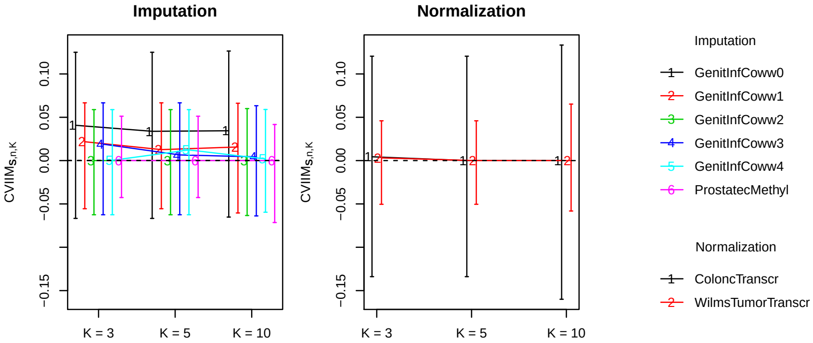
Results: Variable selection and variable filtering by variance



Results: Optimization of tuning parameters



Results: Imputation of missing values and RMA normalization



Simulation results

- Considered data preparation step: variable selection

- Simulated data: $n \in \{50, 100\}$, 2000 correlated normally distributed predictors, two different signal strengthes

- Main results:

  1. Relatively high variance of $\mathrm{CVIIM}_{\boldsymbol{s},n,K}$ - lower for smaller $\mathrm{CVIIM}_{P,n,K}$-values
  2. Negligible bias with respect to the true measure values
  3. Choice of $K = 3$ might be preferable over larger values - smaller variance and better assessment of variance achievable

Summary & Conclusion

- Data preparation steps very often done before CV
  ↯ violation of the separation of training and test data.
  ⇒ Over-optimistic conclusions possible

- Impact very different for different steps - in our analyses greater for steps taking the target variable into account - variable selection and tuning - but not necessarily the case

- New measure CVIIM to assess this impact

- Constantly arising new types of molecular data will require specialized data preparation steps, for which the impact of CV incompleteness will have to be assessed.

Thank you for your attention!

References

Ambroise, C. and McLachlan, G. J. (2002).
Selection bias in gene extraction on the basis of microarray gene-expression data.
Proc. Nat. Acad. Sci. **99,** 6562–6566.

Kostka, D. and Spang, R. (2008).
Microarray based diagnosis profits from better documentation of gene expression signatures.
PLoS Computational Biology **4,** e22.

Simon, R., Radmacher, M. D., Dobbin, K., and McShane, L. M. (2003).
Pitfalls in the use of dna microarray data for diagnostic and prognostic classification.
Journal of the National Cancer Institute **95,** 14–18.

References

Varma, S. and Simon, R. (2006).
Bias in error estimation when using cross-validation for model selection.
BMC Bioinformatics **7,** 91.

Technical Report available online:

Roman Hornung, Christoph Bernau, Caroline Truntzer, Thomas Stadler, and Anne-Laure Boulesteix (2014).
Full versus incomplete cross-validation: measuring the impact of imperfect separation between training and test sets in prediction error estimation.
Department of Statistics, LMU, Technical Report **159**.