

RESEARCH ARTICLE

Optimal ratio for data splitting

V. Roshan Joseph 

H. Milton Stewart School of Industrial and Systems Engineering, Georgia Institute of Technology, Atlanta, Georgia, USA

Correspondence

V. Roshan Joseph, H. Milton Stewart School of Industrial and Systems Engineering, Georgia Institute of Technology, Atlanta, Georgia, USA.
Email: roshan@gatech.edu

Funding information

Division of Chemical, Bioengineering, Environmental, and Transport Systems, Grant/Award Number: DMREF-1921873; Division of Civil, Mechanical and Manufacturing Innovation, Grant/Award Number: 1921646

Abstract

It is common to split a dataset into training and testing sets before fitting a statistical or machine learning model. However, there is no clear guidance on how much data should be used for training and testing. In this article, we show that the optimal training/testing splitting ratio is $\sqrt{p} : 1$, where p is the number of parameters in a linear regression model that explains the data well.

KEYWORDS

testing, training, validation

1 | INTRODUCTION

Data splitting is a commonly used approach for model validation, where we split a given dataset into two disjoint sets: training and testing. The statistical and machine learning models are then fitted on the training set and validated using the testing set. By holding out a set of data for validation separate from training, we can evaluate and compare the predictive performance of different models without worrying about possible overfitting on the training set.

Random subsampling is the most commonly used approach for data splitting, that is, randomly sampling without replacing some rows of the dataset for testing and keeping the rest for training. Deterministic methods for splitting are also proposed in the literature that try to spread out the testing set so that it covers the region spanned by the original dataset in a much better way than a random testing set. CADEX [1], DUPLEX [2], SPXY [3], and SPlit [4] are a few examples of such deterministic methods.

The foregoing data splitting methods can be implemented once we specify a splitting ratio. A commonly used ratio is 80:20, which means 80% of the data is for training and 20% for testing. Other ratios such as 70:30, 60:40, and even 50:50 are also used in practice. There does not seem to be clear guidance on what ratio is best or optimal for a given dataset. The 80:20 split draws its justification from the well-known Pareto principle, but that is again just a thumb rule used by practitioners.

Theoretical and numerical investigations on the optimality of data splitting ratio so far have not led to any consensus. Picard and Berk [5] have recommended 25%–50% for the testing set, whereas Afendras and Markatou [6] recommended 50%. The asymptotic analysis of Larsen and Goutte [7] and Dubbs [8] show that this ratio should get close to 100% as the size of the data becomes very large. On the other hand, extensive numerical studies by Dobbin and Simon [9], Pham et al. [10], and Nguyen et al. [11] have indicated that a value of around 30% to be a reasonable choice.

This article delves into the question of the optimal ratio for data splitting. We propose a new criterion for evaluating the choice of splitting ratio in the next section. Based on this new criterion, we derive a simple closed-form solution for the optimal ratio, which seems to agree with intuition and common practice. Furthermore, a practical strategy to compute the optimal ratio for a given dataset is also proposed.

2 | OPTIMAL RATIO

2.1 | Mathematical formulation

Suppose we have N rows in the dataset that needs to be split into a training set of n rows and testing set of m rows, where $N = n + m$. Let $\gamma = m/N$ denote the splitting ratio. Our aim is to find the optimal γ for a given dataset.

Let $D^{\text{train}} = \{(\mathbf{x}_i, y_i)\}$, $i = 1, \dots, n$ be the training set and $D^{\text{test}} = \{(\mathbf{u}_i, v_i)\}$, $i = 1, \dots, m$ the testing set, where the predictor variables $\mathbf{x}, \mathbf{u} \in \mathbb{R}^d$. If any of the predictor variables is categorical, we assume that they are already converted to numerical variables using some coding technique [4]. Our ultimate aim is to fit a model $g(\mathbf{x}; \boldsymbol{\beta})$, which is expected to approximate the conditional expectation $E(y|\mathbf{x})$, where $\boldsymbol{\beta}$ is a set of unknown parameters in the model. We will use the training set for the estimation of $\boldsymbol{\beta}$, and then evaluate the approximation error using the testing set.

Let $L(y, g(\mathbf{x}; \hat{\boldsymbol{\beta}}))$ be a loss function to assess the approximation/prediction error of the estimated model $g(\mathbf{x}; \hat{\boldsymbol{\beta}})$ from the training set. Then the model's generalization error is given by ([12], Ch. 7)

$$\mathcal{E} = E \left\{ L(y, g(\mathbf{x}; \hat{\boldsymbol{\beta}})) | D^{\text{train}} \right\}, \quad (1)$$

where the expectation is taken with respect to a new realization (\mathbf{x}, y) .

Assume that each row in the dataset is an independent realization from a distribution. If the rows of the testing set $\{(\mathbf{u}_i, v_i)\}_{i=1}^m$ can also be assumed to be from the same distribution, then \mathcal{E} can be estimated using

$$\hat{\mathcal{E}} = \frac{1}{m} \sum_{i=1}^m L(v_i, g(\mathbf{u}_i; \hat{\boldsymbol{\beta}})). \quad (2)$$

If random sampling is used for obtaining the testing set for a given m , then $\text{var}\{\hat{\mathcal{E}}\}$ is of the order $\mathcal{O}(1/m)$. Joseph and Vakayil [4] showed that this variance can be reduced in practice to almost $\mathcal{O}(1/m^2)$ by using support points [13], which is the basis of the SPLIT method.

Now let us turn to the question of what is the optimal m (or γ) to use. If we just focus on the variance of $\hat{\mathcal{E}}$, then a larger m might be the solution. However, a larger m can

lead to poor model fitting and therefore, large values of $\hat{\mathcal{E}}$. Thus it makes sense to split the dataset so that we have a small generalization error with minimum variability. This can be achieved by

$$\min_{\gamma} E \left\{ \hat{\mathcal{E}}^2 \right\}, \quad (3)$$

where the expectation is taken with respect to everything that is random including the training set. Since $E \left\{ \hat{\mathcal{E}}^2 \right\} = E^2\{\hat{\mathcal{E}}\} + \text{var}\{\hat{\mathcal{E}}\}$, (3) will not only minimize the variability of the generalization error, but also its mean.

2.2 | Linear Regression

The criterion in (3) depends on the choice of the model $g(\mathbf{x}; \hat{\boldsymbol{\beta}})$ and the loss function $L(\cdot, \cdot)$. To make the optimization mathematically tractable, we will consider a special case: a linear regression model with squared error loss function.

Let $\mathbf{f}(\mathbf{x}) = (f_1(\mathbf{x}), \dots, f_p(\mathbf{x}))'$ be the set of p features formed based on the d predictor variables, which can include quadratic terms, interaction terms, or other basis functions. To make the notations simple, we will also include the intercept term as part of the features, that is, $f_1(\mathbf{x}) = 1$. Then, the model we would like to fit is

$$g(\mathbf{x}; \boldsymbol{\beta}) = \mathbf{f}(\mathbf{x})' \boldsymbol{\beta}. \quad (4)$$

Let \mathbf{F}_x be the model matrix formed using the training set, that is, the i th row of \mathbf{F}_x is given by $\mathbf{f}(\mathbf{x}_i)'$. Assume the $\text{rank}\{\mathbf{F}_x\} = p \leq n$. Then by minimizing

$$\frac{1}{n} \sum_{i=1}^n \{y_i - \mathbf{f}(\mathbf{x}_i)' \boldsymbol{\beta}\}^2,$$

with respect to $\boldsymbol{\beta}$ we obtain the familiar solution $\hat{\boldsymbol{\beta}} = (\mathbf{F}_x' \mathbf{F}_x)^{-1} \mathbf{F}_x' \mathbf{y}$, where $\mathbf{y} = (y_1, \dots, y_n)'$. Now the generalization error can be estimated from the training set as

$$\hat{\mathcal{E}} = \frac{1}{m} \sum_{i=1}^m \{v_i - \mathbf{f}(\mathbf{u}_i)' \hat{\boldsymbol{\beta}}\}^2.$$

To compute the criterion in (3), we need to make a few assumptions. First, assume that $\{(\mathbf{x}_i, y_i)\}_{i=1}^n$ and $\{(\mathbf{u}_i, v_i)\}_{i=1}^m$ are two independent draws from the same distribution. Note that although each row of the dataset is an independent realization from the distribution, the rows of training and testing sets can become dependent depending on how we split the dataset. Since random sampling and SPLIT maintain the distribution, the independence is a reasonable assumption, but not for the other deterministic

splitting procedures such as CADEX, DUPLEX, and SPXY. Second, assume that

$$E(y|\mathbf{x}) = \mathbf{f}(\mathbf{x})' \boldsymbol{\beta}. \quad (5)$$

In reality, this assumption may not be true, but in the next section, we will explain how to achieve this approximately. Third, assume that $\text{var}(y|\mathbf{x}) = \sigma^2$. Then, we have the following result, whose proof is given in Appendix A.

Proposition 1. Let $\mathbf{A} = \frac{n}{m} (\mathbf{F}_x' \mathbf{F}_x)^{-1} \mathbf{F}_u' \mathbf{F}_u$. Then

$$E\{\hat{\mathcal{E}}\} = \sigma^2 \left\{ 1 + \frac{1}{n} E_X\{\text{tr}(\mathbf{A})\} \right\}, \quad (6)$$

$$\begin{aligned} \text{var}\{\hat{\mathcal{E}}\} = \sigma^4 \left\{ \frac{2}{m} + \frac{4}{mn} E_X\{\text{tr}(\mathbf{A})\} + \frac{2}{n^2} E_X\{\text{tr}(\mathbf{A}^2)\} \right. \\ \left. + \frac{1}{n^2} \text{var}_X\{\text{tr}(\mathbf{A})\} \right\}, \end{aligned} \quad (7)$$

where E_X and var_X denote the expectation and variance taken with respect to the distribution of the predictor variables.

If

$$\frac{1}{n} \mathbf{F}_x' \mathbf{F}_x = \frac{1}{m} \mathbf{F}_u' \mathbf{F}_u, \quad (8)$$

then $\mathbf{A} = \mathbf{I}_p$ and therefore

$$\begin{aligned} E\{\hat{\mathcal{E}}^2\} &= \sigma^4 \left\{ \left(1 + \frac{p}{n} \right)^2 + \frac{2}{m} + \frac{4p}{mn} + \frac{2p}{n^2} \right\} \\ &= \sigma^4 \left\{ 1 + \frac{2p}{N(1-\gamma)} + \frac{2}{N\gamma} \right\} \\ &\quad + \frac{\sigma^4}{N^2} \left\{ \frac{p^2 + 2p}{(1-\gamma)^2} + \frac{4p}{\gamma(1-\gamma)} \right\}. \end{aligned} \quad (9)$$

Picard and Berk [5] suggested splitting the data so that condition (8) holds exactly and obtained the same result using a different criterion. The “matched split” condition in (8) is quite reasonable and is approximately achieved in random subsampling by the law of large numbers. In fact, the approximation gets much better if we were to use SPlit, which minimizes the energy distance [14] between the training and testing sets [15]. The following asymptotic result based on Afendras and Markatou [6] sidesteps the “matched split” requirement.

Proposition 2. If p is fixed and $\lim_{n \rightarrow \infty} \frac{1}{n} \mathbf{F}_x' \mathbf{F}_x = \lim_{m \rightarrow \infty} \frac{1}{m} \mathbf{F}_u' \mathbf{F}_u = \Sigma$ is finite and positive definite, then

$$E\{\hat{\mathcal{E}}^2\} = \sigma^4 \left\{ 1 + \frac{2p}{N(1-\gamma)} + \frac{2}{N\gamma} \right\} + \mathcal{O}\left(\frac{1}{N^2}\right). \quad (10)$$

The proof of Proposition 2 is omitted because it directly follows from Proposition 4 of Afendras and Markatou [6]. Thus, for fixed p and large N , the optimal splitting ratio can

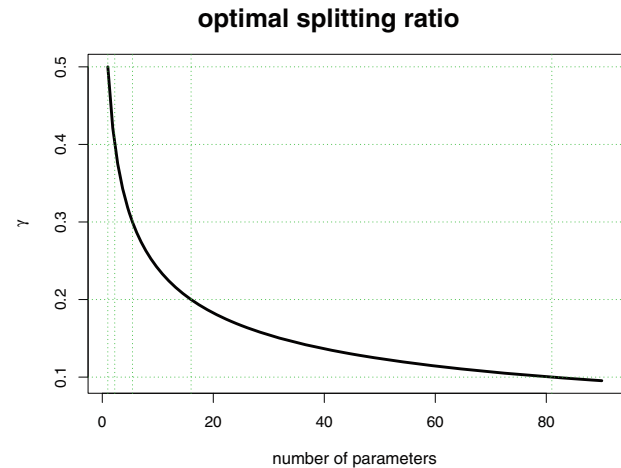


FIGURE 1 Optimal splitting ratio against the number of parameters

be obtained by minimizing

$$\frac{2p}{(1-\gamma)} + \frac{2}{\gamma}.$$

This is minimized at

$$\gamma^* = \frac{1}{\sqrt{p} + 1}, \quad (11)$$

which gives us the main result of this article that we should split the dataset into training and testing using the ratio $\sqrt{p} : 1$. This is plotted in Figure 1. We can see that it gives values in a range that is commonly used in practice. The curve starts at $\gamma^* = 0.5$ when there is only a single parameter to estimate and then decreases to $\gamma^* = 0.1$ when the number of parameters reaches 81. This behavior makes complete sense because we need more training data when there are more parameters to estimate in the model. When the number of parameters p is unknown, the choice of $\gamma = 1/4$ or $\gamma = 1/3$ seems reasonable as recommended by several practitioners. In the next section we will propose a data-driven strategy to estimate p so that we can make a more informative choice of the splitting ratio.

Sometimes it is necessary to split the training set also into two parts for estimating the regularization parameters in the model. In such a case the three sets will be called training, validation, and testing ([12], Ch. 7). Following the optimal ratio, the split sizes for the three sets should be $\{(1-\gamma^*)^2, \gamma^*(1-\gamma^*), \gamma^*\}$. Thus, they should be split according to the ratio $p : \sqrt{p} : (\sqrt{p} + 1)$. For example, if $p = 16$, then the splitting ratio would be $64 : 16 : 20$.

2.3 | Simulations

Since several assumptions have gone into the derivation of the optimal ratio, it makes sense to check the results

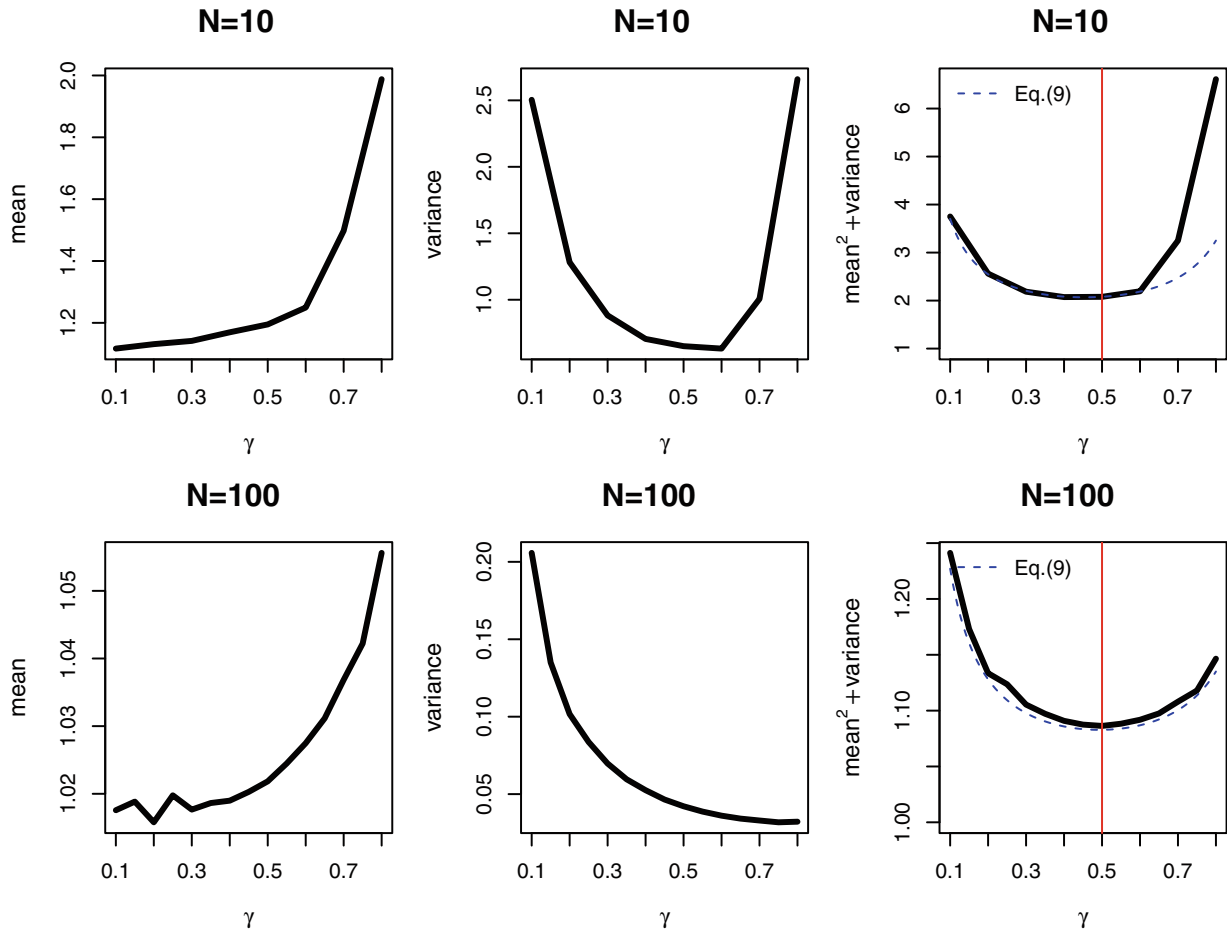


FIGURE 2 Plots of $E\{\hat{\epsilon}\}$, $var\{\hat{\epsilon}\}$, and $E\{\hat{\epsilon}^2\}$ against the splitting ratio for a model with only an intercept ($p = 1$). The top panels shows the plots with $N = 10$ and the bottom panels with $N = 100$

using simulations. First, consider the simplest case with no predictor variables. We generate the data $Y_i \sim \text{iid } \mathcal{N}(\mu, \sigma^2)$ for $i = 1, \dots, N$, and randomly split them into training and testing sets for a given splitting ratio γ . We then estimate μ by \bar{y} from the training set and compute $\hat{\epsilon} = \sum_{i=1}^m (v_i - \bar{y})^2 / m$ using the testing set. This is repeated 10,000 times, and the following quantities are estimated: $E\{\hat{\epsilon}\}$, $var\{\hat{\epsilon}\}$, and $E\{\hat{\epsilon}^2\}$. They are plotted in Figure 2 for different values of γ and for two cases: $N = 10$ and $N = 100$.

We can see from Figure 2 that $E\{\hat{\epsilon}\}$ increases with γ , whereas $var\{\hat{\epsilon}\}$ decreases and then increases. This non-monotonic behavior of the variance might prompt us to find the optimal ratio by simply minimizing the variance. However, such an optimum can drift to 1 as $N \rightarrow \infty$. On the other hand, the proposed criterion $E\{\hat{\epsilon}^2\}$ is well-behaved, which has a clear minimum at $\gamma^* = 0.5$. Although (11) is true only asymptotically as $N \rightarrow \infty$, the approximation seems to be good even for N as small as 10.

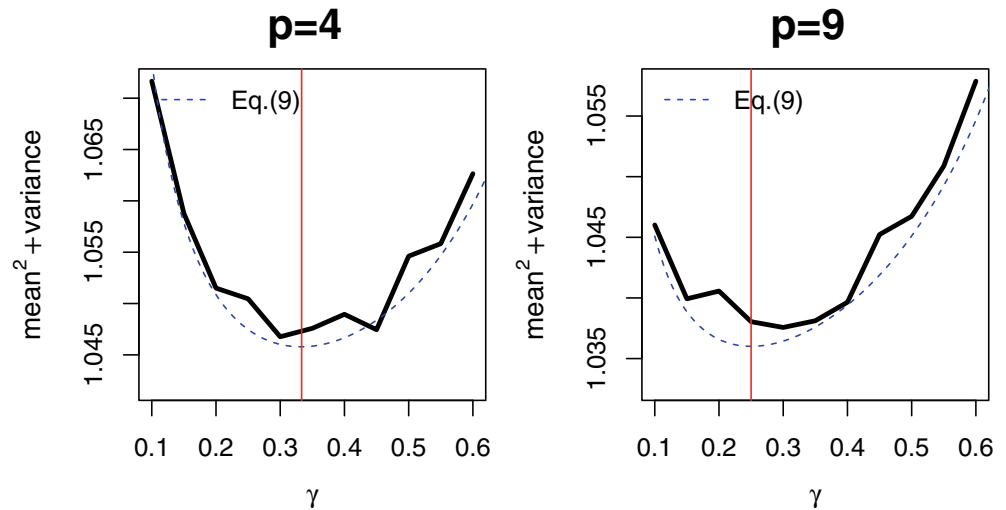
Now consider a polynomial regression model with one predictor and degree $p - 1$: $Y_i = \beta_0 + \beta_1 f_1(x_i) + \dots +$

$\beta_{p-1} f_{p-1}(x_i) + \epsilon_i$, where $f_r(x)$ is a Chebyshev polynomial of degree r and $\epsilon_i \sim \text{iid } \mathcal{N}(0, \sigma^2)$ for $i = 1, \dots, N$. Let $\{x_i\}_{i=1}^N$ be the N Chebyshev nodes and $\sigma^2 = 1$. As before, we estimate $E\{\hat{\epsilon}^2\}$ for various values of γ using 10,000 simulations with $N = 100p$. Figure 3 plot two cases with $p = 4$ and $p = 9$. We can see that the minimum of $E\{\hat{\epsilon}^2\}$ is achieved around γ^* in (11) up to the simulation error, confirming the validity of the theoretical result. The $E\{\hat{\epsilon}^2\}$ in (9) is also plotted in Figure 3, which agrees approximately with the simulation result. This is expected because the “matched split” condition will be approximately achieved with random subsampling of the dataset.

3 | A PRACTICAL STRATEGY

The optimal ratio in (11) is derived under the assumption that $E(y|\mathbf{x}) = \mathbf{f}(\mathbf{x})'\boldsymbol{\beta}$. In reality, this need not be true. In fact, even if the true model is a linear regression model, we may not even know which features of the data and how many of them should be used in the model. Thus, we

FIGURE 3 Plots of $E\{\hat{\mathcal{E}}^2\}$ against γ for a polynomial regression model with $p = 4$ (left) and $p = 9$ (right). The optimal ratio in (11) is shown as a vertical line



need a more practical strategy for deciding on a splitting ratio.

We propose a two-step procedure:

1. Expand the given set of predictor variables $\mathbf{x} = (x_1, \dots, x_d)'$ into a large number of features $f_1(\mathbf{x}), \dots, f_k(\mathbf{x})$ and fit a linear regression model on the *full* data. Use a model selection criterion such as AIC [16] to identify the “true” regression model. This gives a $p \leq k$.
2. Use the p identified in step 1 to compute the optimal ratio in (11) and split the dataset into training and testing sets.

The main assumption in this procedure is that the “true model” can be well approximated by a linear regression model once we expand the feature set. It is important to mention that these features should be independent of data, otherwise we may mistakenly end up choosing a small value for p .

The foregoing procedure can become computationally expensive for large N and k . Therefore, we may consider a simpler approach to find a reasonable value of p . First note that for $k \geq N$, step 1 of the procedure will always find a value of p in the range $[1, N_u]$, where N_u is the number of unique rows in the input matrix of the dataset. Since the regression model is likely to be sparse, the geometric mean of 1 and N_u could be a good choice for p . Therefore, letting $p = \sqrt{N_u}$, (11) becomes

$$\gamma^* = \frac{1}{N_u^{1/4} + 1}. \quad (12)$$

In fact, this solution is a good approximation to the minimizer of (9) for large N even when $p = \sqrt{N_u}$.

We will illustrate the proposed approach using a real dataset. Consider the concrete compressive strength dataset from Yeh [17] which can be obtained from the

UCI Machine Learning Repository [18]. This dataset has $N = 1,030$ rows with eight continuous predictors pertaining to the concrete’s ingredients and age, and one response: concrete’s compressive strength. We create the feature set by including the main effects, two-factor interactions, and quadratic terms of the eight predictors. Stepwise regression using AIC gave a model with $p = 40$ features including the intercept. Using (11), we obtain $\gamma^* = 0.1365$. On the other hand, (12) gives $\gamma^* = 0.1512$, which is in good agreement with the value obtained using variable selection.

To get a reliable answer with a few simulations, we use SPlit (using R package SPlit [19]) to split the data into training and testing sets in the ratio 85:15. Four models are fitted on the training set: (1) Lasso including quadratic and two-factor interaction terms (using R package glmnet [20]), (2) Random Forest (using R package randomForest [21]), (3) Kernel Ridge Regression (using R package listdtr [22]), and (4) Gaussian process regression (using R package laGP [23]). The root mean squared prediction error is then computed on the testing set (i.e., $\sqrt{\hat{\mathcal{E}}}$). This procedure is repeated 30 times and the results are plotted in the left panel of Figure 4.

We can see that Kernel Ridge Regression and Gaussian process regression give the best results. In fact, both use separable Gaussian kernels and therefore, they are the same except that Kernel Ridge Regression is tuned using cross-validation, whereas Gaussian process regression is tuned using maximum likelihood. Kernel Ridge Regression seems to be slightly better than Gaussian process regression in this particular problem because it gives low prediction errors in most of the cases. The large variability of the root mean squared errors of Kernel Ridge Regression could be due to an issue with the convergence of the optimization method used for tuning the parameters. So, if special care is taken to tune it, this method should perform the best in future scenarios among the

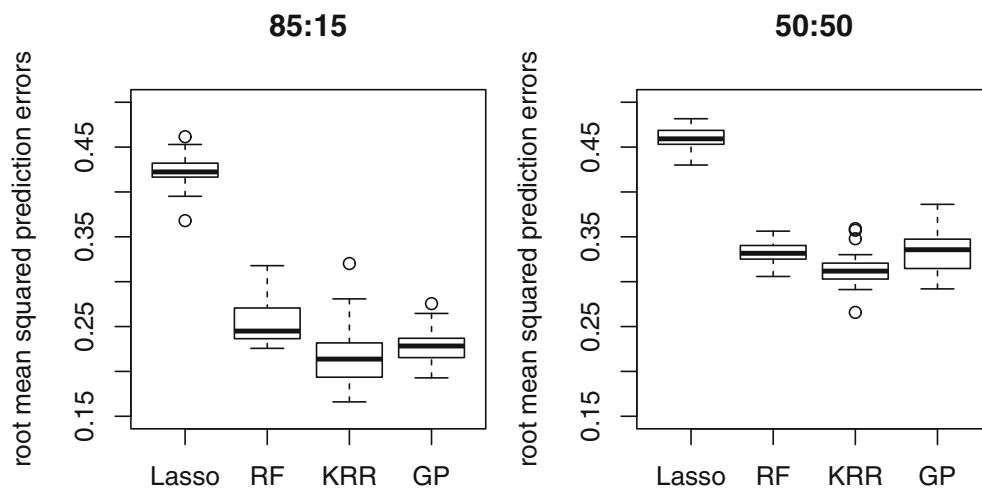


FIGURE 4 Boxplot of root mean squared prediction errors of different modeling methods (Lasso, Random Forest, Kernel Ridge Regression, and Gaussian Process) in the concrete dataset with 85:15 split (left) and 50:50 split (right)

four methods considered in this study. These conclusions would not have been possible to make without using data splitting.

For comparison, we have also shown the results with a 50:50 split in the right panel of the same figure. We can see that the root mean squared prediction errors of all the four methods are larger in this case, but our selection of the winner will not be affected as the Kernel Ridge Regression still seem to be doing better than the other three. However, the random forest now seems to be better than Gaussian process regression, which was not the case before. Thus, the ranking of the methods can change with the splitting ratio, and therefore, it would be better to examine the plot at the optimal splitting ratio to arrive at a more reliable result.

4 | CONCLUSIONS

In this article we have shown that a dataset should be split in the ratio $\sqrt{p} : 1$ for creating training and testing sets, where p is the number of parameters to estimate in a linear regression model that fits the data well. We have also discussed a practical strategy to find p using variable selection methods on the full dataset with an expanded feature set. We also found $p = \sqrt{N_u}$ to be a reasonable choice, where N_u is the number of unique rows in the input matrix of the dataset.

Linear regression is used in this work only as a tool for approximately finding the “true” model and therefore, the results can be used for selecting the best method among different modeling choices. When the input–output relationship is complex, it is difficult or even impossible to identify the true model using linear regression, but the hope is that the result will be at least better than choosing a splitting ratio blindly. Although we have derived the

optimal splitting ratio for regression problems, we believe that it can also serve as a guideline for classification purposes because fitting a logistic regression model can be viewed as fitting a linear regression model to some continuous latent data. However, more research is necessary to understand its usefulness in classification problems.

In this new era of data science, analysts are fitting models with thousands of parameters, which might suggest that we should keep most of the data for training. However, even if the model contains a large number of parameters, they are estimated with a high amount of regularization, and therefore the effective number of parameters [24] could be small. Thus the strategy given here for finding optimal data splitting ratio could still work.

Another scenario that one might encounter is the availability of physics-based models where the models may contain only a few parameters such as rate constants in a chemical kinetics model. The linear regression-based strategy proposed here might suggest a much larger p as many basis functions might be needed to fully capture the response surface. However, physics-based models are derived under some simplifying assumptions, and therefore, to fully validate such models we might need to estimate its discrepancy using nonparametric [25] or parametric [26] regression models. Thus, the strategy introduced here based on linear regression methodology is still applicable.


ACKNOWLEDGMENTS

This research is supported by U.S. National Science Foundation grants and CMMI-1921646 and DMREF-1921873.

DATA AVAILABILITY STATEMENT

The data that support the findings of this study are openly available. The codes are available in the R package Split [19].

ORCID

V. Roshan Joseph  <https://orcid.org/0000-0002-9430-5301>

REFERENCES

1. R. W. Kennard and L. A. Stone, *Computer aided design of experiments*, *Technometrics* 11 (1969), no. 1, 137–148.
2. R. D. Snedecor, *Validation of regression models: Methods and examples*, *Technometrics* 19 (1977), no. 4, 415–428.
3. R. K. H. Galvão, M. C. U. Araujo, G. E. José, M. J. C. Pontes, E. C. Silva, and T. C. B. Saldanha, *A method for calibration and validation subset partitioning*, *Talanta* 67 (2005), no. 4, 736–740.
4. V. R. Joseph and A. Vakayil, *SPLIT: An optimal method for data splitting*, *Technometrics* (2021). <https://doi.org/10.1080/00401706.2021.1921037>.
5. R. R. Picard and K. N. Berk, *Data splitting*, *Am. Statist.* 44 (1990), no. 2, 140–147.
6. G. Afendras and M. Markatou, *Optimality of training/test size and resampling effectiveness in cross-validation*, *J. Statist. Plan. Infer.* 199 (2019), 286–301.
7. J. Larsen and C. Goutte, “On optimal data split for generalization estimation and model selection,” *Neural networks for signal processing IX: Proceedings of the 1999 IEEE signal processing society workshop (Cat. No. 98TH8468)*, IEEE, 1999, pp. 225–234.
8. Dubbs, A. (2021). Test set sizing via random matrix theory. arXiv preprint arXiv:2112.05977.
9. K. K. Dobbin and R. M. Simon, *Optimally splitting cases for training and testing high dimensional classifiers*, *BMC Med. Genet.* 4 (2011), no. 1, 1–8.
10. B. T. Pham, I. Prakash, A. Jaafari, and D. T. Bui, *Spatial prediction of rainfall-induced landslides using aggregating one-dependence estimators classifier*, *J. Indian Soc. Remote Sens.* 46 (2018), no. 9, 1457–1470.
11. Q. H. Nguyen, H.-B. Ly, L. S. Ho, N. Al-Ansari, H. V. Le, V. Q. Tran, I. Prakash, and B. T. Pham, *Influence of data splitting on performance of machine learning models in prediction of shear strength of soil*, *Math. Probl. Eng.* 2021 (2021).
12. T. Hastie, R. Tibshirani, and J. Friedman, *The elements of statistical learning: Data mining, inference, and prediction*, Springer, New York, 2009.
13. S. Mak and V. R. Joseph, *Support points*, *Ann. Stat.* 46 (2018), no. 6A, 2562–2592.
14. G. J. Székely and M. L. Rizzo, *Energy statistics: A class of statistics based on distances*, *J. Statist. Plan. Inf.* 143 (2013), no. 8, 1249–1272.
15. Vakayil, A. and Joseph, V. R. (2022). Data twinning. *Statistical Analysis and Data Mining: The ASA Data Science Journal*, <https://doi.org/10.1002/sam.11574>.
16. Akaike, H. (1973). Information theory and an extension of the maximum likelihood principle. In *Proceedings of the 2nd international symposium on information theory* (eds B. N. Petrov and F. Csaki), pages 267–281. Akademiai Kiado, Budapest.
17. I.-C. Yeh, *Modeling of strength of high-performance concrete using artificial neural networks*, *Cem. Concr. Res.* 28 (1998), no. 12, 1797–1808.
18. Dua, D. and Graff, C. (2017). UCI Machine Learning Repository. <http://archive.ics.uci.edu/ml>.
19. Vakayil, A., Joseph, V. R., and Mak, S. (2022). SPLIT: Split a Dataset for Training and Testing. R package version 1.2.
20. J. Friedman, T. Hastie, and R. Tibshirani, *Regularization paths for generalized linear models via coordinate descent*, *J. Stat. Softw.* 33 (2010), no. 1, 1–22.
21. A. Liaw and M. Wiener, *Classification and regression by random forest*, *R News* 2 (2002), no. 3, 18–22.
22. Zhang, Y. (2021). listdr: List-based rules for dynamic treatment regimes. R 1.1.
23. R. B. Gramacy, *laGP: Large-scale spatial modeling via local approximate gaussian processes in R*, *J. Stat. Softw.* 72 (2016), no. 1, 1–46.
24. J. Moody, *The effective number of parameters: An analysis of generalization and regularization in nonlinear learning systems*, *Adv. Neur. Inform. Process. Syst.* 4 (1991), 847–854.
25. M. C. Kennedy and A. O’Hagan, *Bayesian calibration of computer models*, *J. R. Statist. Soc. Series B (Statist. Methodol.)* 63 (2001), no. 3, 425–464.
26. V. R. Joseph and S. N. Melkote, *Statistical adjustments to engineering models*, *J. Qual. Technol.* 41 (2009), no. 4, 362–375.

How to cite this article: V. R. Joseph, *Optimal ratio for data splitting*, *Stat. Anal. Data Min.: ASA Data Sci. J.* **15** (2022), 531–538. <https://doi.org/10.1002/sam.11583>

APPENDIX A. PROOF OF PROPOSITION 1

Let \mathbf{X} be the data corresponding to the predictor variables in the dataset. Since $E(\hat{\beta}|\mathbf{X}) = \beta$ and $\text{var}(\hat{\beta}|\mathbf{X}) = \sigma^2(\mathbf{F}_x'\mathbf{F}_x)^{-1}$

$$\begin{aligned}
 E(\hat{\mathcal{E}}|\mathbf{X}) &= \frac{1}{m} \sum_{i=1}^m E \left\{ \left(v_i - f(\mathbf{u}_i)' \hat{\beta} \right)^2 | \mathbf{X} \right\} \\
 &= \frac{1}{m} \sum_{i=1}^m \left\{ \left(f(\mathbf{u}_i)' \beta - f(\mathbf{u}_i)' \beta \right)^2 \right. \\
 &\quad \left. + \sigma^2 + \sigma^2 f(\mathbf{u}_i)' (\mathbf{F}_x' \mathbf{F}_x)^{-1} f(\mathbf{u}_i) \right\} \\
 &= \sigma^2 + \frac{1}{m} \sigma^2 \text{tr} \left\{ \mathbf{F}_u (\mathbf{F}_x' \mathbf{F}_x)^{-1} \mathbf{F}_u' \right\} \\
 &= \sigma^2 \left\{ 1 + \frac{1}{n} \text{tr}(\mathbf{A}) \right\}.
 \end{aligned}$$

Thus we obtain (6) using $E(\hat{\mathcal{E}}) = E_x[E(\hat{\mathcal{E}}|\mathbf{X})]$. Now consider the conditional variance:

$$\begin{aligned}
 \text{var}(\hat{\mathcal{E}}|\mathbf{X}) &= E[\text{var}(\hat{\mathcal{E}} | \mathbf{X}, \mathbf{y}) | \mathbf{X}] + \text{var}[E(\hat{\mathcal{E}} | \mathbf{X}, \mathbf{y}) | \mathbf{X}] \\
 &= \frac{1}{m^2} E \left[\sum_{i=1}^m \text{var} \left\{ \left(v_i - f(\mathbf{u}_i)' \hat{\beta} \right)^2 | \mathbf{X}, \mathbf{y} \right\} \right]
 \end{aligned}$$

$$\begin{aligned}
& + \frac{1}{m^2} \text{var} \left\{ \sum_{i=1}^m \left(\mathbf{f}(\mathbf{u}_i)' \boldsymbol{\beta} - \mathbf{f}(\mathbf{u}_i)' \hat{\boldsymbol{\beta}} \right)^2 | \mathbf{X} \right\} \\
& = \frac{4\sigma^2}{m^2} E \left\{ (\hat{\boldsymbol{\beta}} - \boldsymbol{\beta})' \mathbf{F}_u' \mathbf{F}_u (\hat{\boldsymbol{\beta}} - \boldsymbol{\beta}) | \mathbf{X} \right\} + \frac{2\sigma^4}{m} \\
& \quad + \frac{1}{m^2} \text{var} \left\{ (\hat{\boldsymbol{\beta}} - \boldsymbol{\beta})' \mathbf{F}_u' \mathbf{F}_u (\hat{\boldsymbol{\beta}} - \boldsymbol{\beta}) | \mathbf{X} \right\} \\
& = \frac{4\sigma^4}{m^2} \text{tr} \left(\mathbf{F}_u' \mathbf{F}_u (\mathbf{F}_x' \mathbf{F}_x)^{-1} \right) + \frac{2\sigma^4}{m} \\
& \quad + \frac{2\sigma^4}{m^2} \text{tr} \left(\mathbf{F}_u' \mathbf{F}_u (\mathbf{F}_x' \mathbf{F}_x)^{-1} \mathbf{F}_u' \mathbf{F}_u (\mathbf{F}_x' \mathbf{F}_x)^{-1} \right).
\end{aligned}$$

Thus, (7) follows from the identity $\text{var}(\hat{\mathcal{E}}) = E_{\mathbf{X}}\{\text{var}(\hat{\mathcal{E}}|\mathbf{X})\} + \text{var}_{\mathbf{X}}\{E(\hat{\mathcal{E}}|\mathbf{X})\}$.