Representative random sampling: an empirical evaluation of a novel bin stratification method for model performance estimation

Michael C. Rendleman¹,³ · Brian J. Smith² · Guadalupe Canahuate³ · Terry A. Braun¹,⁴ · John M. Buatti⁵ · Thomas L. Casavant¹,³,⁴

Received: 15 June 2022 / Accepted: 27 August 2022 / Published online: 27 October 2022 © The Author(s) 2022

Abstract

High-dimensional cancer data can be burdensome to analyze, with complex relationships between molecular measurements, clinical diagnostics, and treatment outcomes. Data-driven computational approaches may be key to identifying relationships with potential clinical or research use. To this end, reliable comparison of feature engineering approaches in their ability to support machine learning survival modeling is crucial. With the limited number of cases often present in multi-omics datasets (“big p, little n,” or many features, few subjects), a resampling approach such as cross validation (CV) would provide robust model performance estimates at the cost of flexibility in intermediate assessments and exploration in feature engineering approaches. A holdout (HO) estimation approach, however, would permit this flexibility at the expense of reliability. To provide more reliable HO-based model performance estimates, we propose a novel sampling procedure: representative random sampling (RRS). RRS is a special case of continuous bin stratification which minimizes significant relationships between random HO groupings (or CV folds) and a continuous outcome. Monte Carlo simulations used to evaluate RRS on synthetic molecular data indicated that RRS-based HO (RRHO) yields statistically significant reductions in error and bias when compared with standard HO. Similarly, more consistent reductions are observed with RRS-based CV. While resampling approaches are the ideal choice for performance estimation with limited data, RRHO can enable more reliable exploratory feature engineering than standard HO.

Keywords Sampling · Performance estimation · Machine learning · Modeling · Genomics

1 Introduction

In medicine and biology, the availability and dimensionality of multi-omics data are increasing. Initiatives such as The Cancer Genome Atlas (TCGA) have collected and made available data from more than 20,000 patient cases across more than 30 cohorts of cancer. For specific diseases, there are far fewer cases (n) than there are features (p). This is often
referred to as a “big $p$, little $n$” problem, $p \gg n$. Analyses are further complicated by the right-censored nature of time-to-event data, as there may be cases where death or recurrence occurs after the latest follow-up time.

Modern techniques in machine learning (ML) modeling can be effective tools for knowledge and hypothesis generation in this $p \gg n$ setting, but intelligent decisions in dimensionality reduction and feature engineering (FE) are crucial to their effectiveness (Shi et al 2019; Rendleman et al 2019; Kuhn and Johnson 2020, Chapter 10). To this end, the exploration, tuning, and comparison of FE approaches are necessary.

When evaluating FE techniques for modeling, the best practices involve cross validation (CV) or another resampling technique to estimate generalization (extra-sample) performance. However, resampling limits exploration of novel and experimental FE techniques, as all preprocessing must be designed and parameterized ahead of time to allow systematic application to the training data in each iteration (Kuhn and Johnson 2020, Chapter 3.4.7).

For this reason, we select a holdout (HO) approach, which has lower computational requirements but higher variability in generalization performance estimates. To counteract this variability, we propose a continuous stratified sampling method to ensure that the training and test sets are representative of the full outcome distribution.

Representative random sampling (RRS) is a sampling method applying constrained equipopulous bin stratification, where the observed distribution of the regression (or right-censored) outcome is retained in train/test splits. This is achieved by partitioning the data into equally sized bins based on the outcome variable and selecting HO sets (or CV folds) with stratified sampling on the partitions. The number of data points assigned to each bin is determined by the desired test/train split or the number of CV folds.

While bin stratification is not novel (Kuhn and Johnson 2020, Chapter 3), the current literature lacks empirical analysis of how it affects generalization performance estimation. In this paper, we characterize statistical properties of samplings generated by RRS and employ Monte Carlo simulations to determine how this continuous stratification approach influences error and bias in generalization model performance measurements.

The focus of this article is on right-censored survival outcomes, but this sampling approach can be applied to any time-to-event or continuous regression prediction setting.

2 Representative random sampling (RRS) procedure

In standard CV or HO, group assignment is performed randomly with respect to the outcome variable. With representative random sampling (RRS), a continuous stratified sampling method ensures that each group is representative of the full outcome distribution. Data points are partitioned into bins based on the continuous outcome variable, and stratified sampling is used to assign groupings for HO or CV.

The distinction between RRS and typical bin stratification lies in the definition of the bin. Instead of predefined ranges, the bins reflect the sorted order of data points. Bin boundaries are selected such that each bin contains an equal number of data points. The bin size parameter $k$ is chosen to be as small as possible for the desired number of groupings. This minimum bin size is critical in preventing statistically significant relationships between the selected random grouping and the outcome of interest.

**Algorithm 1** RRS procedure to assign $n$ data points to $k$ groups. Without loss of generality, assume $n$ is a multiple of $k$.

| Require: | $n$ data points $X = \{x_1, x_2, \cdots, x_n\}$ |
|----------|-----------------------------------------------|
| Require: | $X$ sorted on outcome, $x_{i_{\text{min}}} \leq x_{i+1}$ |
| 1: | $i \leftarrow 0$ |
| 2: | Initialize $k$ empty groups $F_1, F_2, \cdots, F_k = \emptyset$ |
| 3: | while $i < n/k$ do |
| 4: | $B = \{X[i \times k + 1], X[i \times k + 2], \cdots, X[i \times k + k]\}$ ⇒ Assign $k$ sequential data points to bin $B$ |
| 5: | for each $j \in \{1, 2, \cdots, k\}$ do |
| 6: | $a \sim B$ ⇒ Randomly draw a data point |
| 7: | $F_j = F_j \cup \{a\}$ ⇒ Assign $a$ to group $F_j$ |
| 8: | $B = B \setminus \{a\}$ ⇒ Remove $a$ from bin $B$ |
| 9: | end for |
| 10: | $i = i + 1$ ⇒ Iterate to next $k$ data points |
| 11: | end while |

In Algorithm 1, $k$ sequential data points are assigned to $k$ separate groups in each iteration. In the case that $n$ is not a multiple of $k$, the remaining $n \mod k$ data points in $X$ can be randomly assigned to unique folds with one additional iteration. This algorithm generates folds for $k$-fold CV. Additionally, groupings generated with this algorithm can be used to assign data points to sets of differing size as in a holdout procedure. For example, for a one-third holdout, set $k = 3$ and randomly assign one group to be the testing set. In this work, RRS-based variants of HO and CV are denoted as RRHO and RRCV, respectively.

For the case of right-censored data, the precision of performance estimates depends on the relative fractions of observed and censored data points. Accordingly, the observed events are evenly distributed throughout the sampled sets by separately assigning the censored and uncensored data points to $k$ groups via RRS and combining the corresponding groups. This ensures that the resulting groupings have both censored and uncensored data in the same proportions found in the full dataset and any distribution differences due to non-random censorship are retained.
3 Methods

3.1 Generalization performance estimation

Holdout (HO) and cross validation (CV) are two common ways to estimate generalization performance of a predictive model using all available data. In HO, the model is trained on some portion of the data, while the rest of the data is “held out” to be tested on after training. When you have a small number of data points, HO estimates of generalization performance can have high variance, as they strongly depend on the data points chosen for training and testing.

CV, on the other hand, is a way of systematically repeating HO such that every data point is used for testing exactly once. The data are partitioned into k folds, and the training procedure is conducted k times. In each iteration, one fold is selected to be the testing data. Afterward, model performance measures from each iteration are averaged.

As noted in Sect. 2, we refer to RRS-based variants of HO and CV as RRHO and RRCV, respectively.

3.2 Survival modeling approaches

Ensemble models combine the predictions of multiple models to produce more accurate, robust, and reliable predictions than single learner models at the cost of interpretability (Ardabili et al 2020). Four ensemble survival modeling approaches are applied in this work: gradient boosting with linear models (GLMBoost), gradient boosted regression trees (BlackBoost), random survival forests (RFSRC), and conditional inference random survival forests (CForest).

The primary distinctions between these approaches are the base learners applied and the method of ensemble construction. GLMBoost fits a series of linear survival models and BlackBoost fits multiple survival trees, but both use a gradient boosting approach, training subsequent models to account for residuals leftover by the other base learners (Hothorn et al 2022). In contrast, RFSRC and CForest both apply a variation of Breiman’s random forest (RF) modified to handle right-censored survival data. With RFSRC, the base learners are standard survival trees (Ishwaran and Kogalur 2022), but with CForest, the models trained are conditional inference survival trees (Hothorn et al 2006; Strobl et al 2007, 2008).

When training models in the Monte Carlo simulations, we used the implementations provided in the MachineShop R package using default hyperparameters (Smith 2021).

3.3 TCGA-HNSC

Our analyses made significant use of TCGA’s head and neck squamous cell carcinoma (HNSCC) dataset, TCGA-HNSC (The Cancer Genome Atlas Network 2015). To investigate statistical properties of RRS-based samplings, right-censored survival times were used. To generate simulated data in the Monte Carlo simulations, both survival times and RNA expression variables were applied indirectly and directly, respectively.

In the survival data, approximately two-thirds of cases are censored. The distributions of censored and observed survival times do not significantly differ, but are still sampled separately with RRS. In the mRNAseq expression data, only the 520 patients with solid-tumor samples are included. Additionally, gene expression features with missing values or normalized variance \( \sigma^2 \leq 0.005 \) were excluded, leaving 16,628 expression features across 520 patient cases.

3.4 Repeated samplings

To investigate the general properties of one-third HO sets selected using RRS, we employed the survival data present in TCGA-HNSC. HO group assignment was performed 10,000 times with several different types of sampling, including standard random sampling, RRS, and equipopulous bin stratification with non-minimum bin sizes.

For RRS-based one-third HO, we set \( k = 3 \). TCGA-HNSC has 528 cases with survival data, giving a maximum bin size of 528/\( k = 176 \). Logrank testing was performed on each group assignment. The distributions of the resulting logrank p-values are reported per sampling approach.

3.5 Monte Carlo simulations

To examine the effects of RRS on generalization performance estimation, the methods of Borra and Di Ciaccio (2010) were adapted. A Monte Carlo simulation is employed to compare 1/3 HO and 10-fold CV with and without RRS.

3.5.1 Simulated data

To provide a rich transcriptomics dataset, 3000 simulated data points are generated using TCGA-HNSC transcriptomics data. For each data point, a total of 1000 simulated survival outcomes are generated such that they depend on the simulated data. Five-hundred of the outcomes used Weibull parameters tuned to match the survival times in TCGA-HNSC, and randomly chosen parameters generated the remaining simulated outcomes. Results for these “TCGA-like” and “General” outcomes are reported separately. Full details of simulated data generation are given in Appendix A.1.

3.5.2 Simulated model performance

The process described in the following paragraph is depicted in Fig. 1. For each simulated outcome, a sample of 500 of the 3000 data points is selected to be the “available” data. Sev-
Fig. 1 Flow chart depicting the procedure for a single step in the Monte Carlo simulations, including simulated data sampling, survival model training, and generalization performance calculation and estimation.

Details on the ensemble modeling approaches and performance metrics are given in Sects. 3.2 and 3.5.3, respectively.

eral types of ML models are trained on the available data, and the remaining 2500 data points are used to calculate the models’ true generalization performance. Then, generalization performance estimation is done on the available data using four approaches: HO, RRHO, CV, and RRCV. With true and estimated generalization performance, the error and bias of the estimates can be calculated and compared (Borra and Di Ciaccio 2010).

For each simulated survival outcome, this process is repeated 30 times with different samples of 500 data points. After aggregating error and bias across all simulated outcomes and samplings (see Appendix A.2), the standard HO and CV approaches are compared to their RRS counterparts using Welch’s two-sample t-test.

3.5.3 Generalization performance metrics

To measure predictive performance of survival models, three metrics are used: mean squared error (MSE), mean absolute error (MAE), and concordance index (C-index). MSE and MAE depend on the scale of the output, measuring the variance and average of the residuals, respectively. In contrast, the C-index is a measure of how effective a model is at relative ordering of outputs; intuitively, it is the probability that the predictions of any two data points are correctly ordered. Notably, MSE and MAE are only defined with uncensored cases, rendering them less reliable when the number of right-censored data points is high. As described in Sect. 3.3, approximately two-thirds of simulated and TCGA-HNSC data points are censored, making the C-index most reliable.

4 Results and discussion

4.1 RRS samplings

The logrank test is typically used to compare time-to-event distributions between populations, where $p_{\text{logrank}} < 0.05$ indicates statistically significant differences. Figure 2 reports the distribution of $p_{\text{logrank}}$ values from logrank tests applied to 10,000 repetitions of each sampling approach. It illustrates how equipopulous bin-stratified sampling restricts the set of possible random samplings to a subset that avoids statistically significant relationships between group assignment and time-to-event outcome.

The strength of this effect depends on the size of the bins used for stratified sampling. As bin size decreases, the $p_{\text{logrank}}$ values associated with test/train groupings increase.
Fig. 2 Distribution of logrank $p$-values for repeated group assignment with RRS 1/3 HO, 1/3 HO using equipopulous bin stratification with non-minimum bin sizes, and a standard 1/3 HO (denoted “random”). A vertical line indicates the significance level $p_{\text{logrank}} = 0.05$.

As expected for a fully random 1/3 HO, approximately 5% of the samplings yield $p_{\text{logrank}} < 0.05$. With a minimum bin size equipopulous bin stratification (RRS) 1/3 HO, no samplings exhibited statistically significant differences. Even in the case of maximum bin size, the percentage of possible samplings with statistically significant differences was reduced to 0.1%.

This shows that RRS, using minimum bin size, maximizes similarity in the distribution of outcomes across HO sets or CV folds selected using equipopulous bin-stratified sampling.

4.2 Monte Carlo simulations

Figure 3 reports the error and bias for C-index estimates across all simulations and model types. The raw data are also provided in Tables 1 and 2 for the error and bias, respectively.

RRS-based estimation variants consistently yielded statistically significant reductions in average relative bias in nearly all cases. Across HO estimation approaches, boosted models exhibited the least bias. Application of RRS to CV estimation reduced the bias of RF-based models to be on par with the least biased model type, GLMBoost.

For average error, RF-type models saw modest but significant decreases with CV. Error reduction with HO approaches was less consistent across the two sets of simulated survival outcomes, but statistical significance was found with both RFSRC and GLMBoost. Overall, GLMBoost gave the most accurate model performance estimates.

When estimating MAE and MSE, statistically significant reductions in error and bias tended to be with the RF-based models. Additionally, RF-type models were more reliable on average than the boosting-based models for these metrics. As previously noted, these metrics are less reliable than C-index in this setting, as they are only defined for uncensored data points.

5 Conclusions

RRS is a continuous stratified sampling technique that minimizes statistically significant relationships between random group assignments and a time-to-event or continuous outcome. As a bin stratification approach, it is unique in applying equipopulous bins of minimum size.

RRS has valuable applications in train/test sampling for model performance evaluation. Simulations show that
Fig. 3  Average error and bias values from Monte Carlo simulations when using C-index to evaluate survival models. Results are reported across multiple models and estimation approaches. Statistical comparisons were made between standard approaches (HO, CV) and their respective RRS-based variants. * indicates that for that pair of methods, differences in mean performance were statistically significant (\( p < 0.05 \)).

Table 1  Relative error (mean rRMSE) when using C-index to evaluate survival models

| Survival model | TCGA-like | General |
|----------------|-----------|---------|
| GLMBoost | 2.240 | 2.239 |
| BlackBoost | 3.027 | 2.912 |
| RFSRC | 2.773 | 2.932 |
| CForest | 2.819 | 2.803 |

Results are separated based on whether simulated data generation used the “TCGA-like” parameters. HO or CV pairs with statistically significant differences in mean performance are bolded.

Table 2  Relative bias (mean arb \( \times 10^{-3} \), for brevity) when using C-index to evaluate survival model types

| Survival model | TCGA-like | General |
|----------------|-----------|---------|
| GLMBoost | 6.12 | 6.78 |
| BlackBoost | 7.76 | 8.10 |
| RFSRC | 15.9 | 14.9 |
| CForest | 10.4 | 10.6 |

Results are separated based on whether simulated data generation used the “TCGA-like” parameters. HO or CV pairs with statistically significant differences in mean performance are bolded.
on average, when compared with standard HO and CV approaches, it gives statistically significant reductions in average error and bias metrics across several types of models and performance metrics. Notably, we did not observe increases in these metrics on average with the application of RRS.

This enables more reliable HO-based model performance estimation, in turn improving the credibility of hypotheses and knowledge generated by model analyses. In "big p, little n" data settings, RRS will allow data-driven FE approaches to be explored via ML modeling with higher confidence. RRS can also improve reliability of k-fold CV performance estimates in contexts where repeated CV is infeasible or k is small.

The logical next steps of this work will involve the application of RRHO to FE problems in cancer genomics. This will facilitate the evaluation of FE methods via HO performance estimates and highlight approaches that are effective at extracting meaningful signal in this p >> n setting. Additional comparisons among RRCV, the number of CV folds, and repeated CV could be fruitful, as they may distinguish other suitable contexts for the application of RRS.

Acknowledgements This research was supported in part by computational resources provided by the University of Iowa, Iowa City, Iowa. Additionally, the results published here make use of data generated by the TCGA Research Network: https://portal.gdc.cancer.gov.

Author Contributions MR designed and implemented the RRS algorithm, the experiments evaluating it, the Monte Carlo simulations, and simulated data. Additionally, he drafted, edited, and prepared the manuscript for submission. BS assisted with design of the experiments, Monte Carlo simulations, and statistical calculations. GC provided guidance in the design of the RRS algorithm and in simulated data generation. TB and JB provided feedback on experimental design and contributed to the experimental design, and assisted with editing the manuscript. All authors read and approved the manuscript.

Funding This work was supported by National Institutes of Health/ National Cancer Institute Grant Number U01CA140206. The funding body played no role in the design of our experiments, analysis and interpretation of data, or the writing of this manuscript.

Data availability The TCGA data analyzed in the current study are available in the National Cancer Institute Genomic Data Commons Data Portal, https://portal.gdc.cancer.gov/

Appendix A

A.1 Simulated data generation

Simulated mRNA expression data points are generated from the TCGA-HNSC mRNA expression data as follows: For each of the 16,628 mRNA expression features, values are rescaled to the range [0,1] and 3000 random samples with replacement are taken. The ith sample is assigned to the ith simulated data point. The same set of 3000 simulated data points is used for all simulations.

Each simulated outcome is constructed by randomly selecting 20 features and assigning 20 corresponding regression coefficients from N(0, 1). For every data point, outcomes were generated using a Weibull survival time generator as follows:

\[ T = \left( \frac{9000}{\lambda e^{\beta x}} \right)^{\frac{1}{\alpha}}. \]  

(A1)

where \( x \) represents a single simulated data point, and \( \beta \) denotes the vector of corresponding regression coefficients.

This Weibull survival time generating function was adapted from the generator described by Bender et al (2005) and expanded on by Austin (2012). The most notable difference is replacement of \( \log(u) \), \( u \in (0, 1) \) with 9000. This term sets the overall scale of the survival times, in this case chosen to yield survival times in the same range as found in the TCGA-HNSC dataset (Bender et al 2005; Austin 2012).

Five-hundred TCGA-like outcomes were generated with shape and scale parameters \( \alpha \) and \( \lambda \) fixed to match the survival distribution present in TCGA-HNSC (\( \alpha = 0.92, \lambda = 2062 \)). These parameters were determined by training a null Weibull model on TCGA-HNSC’s clinical survival data. For the other 500 outcomes, the shape and scale parameters were selected randomly: \( \alpha \) was uniformly sampled from the range [0.5, 4], and \( \lambda \) was uniformly sampled from the integers [10, 3000]. Results for the two sets of outcomes are reported separately. Figure 4 presents several generated survival curves.
Fig. 4 Density histograms of real and simulated survival outcomes for 
a TCGA-HNSC survival data, b two simulations using Weibull parameters derived from TCGA-HNSC (TCGA-like), and c two simulated survival outcomes generated using randomly selected Weibull parameters (general). Weibull parameters ($\alpha$, $\lambda$) are reported for each simulated outcome.
A.2 Error and Bias calculations

In the Monte Carlo simulations, the relative error and relative bias of different generalization performance estimation approaches were calculated for each combination of simulation parameters (survival model type and performance metric). These calculations are based on the comparison of bootstrap sampling and CV techniques in Borra and Di Ciaccio (2010).

For each outcome $h$ and set of simulation parameters, relative error was measured using relative root-mean-squared error ($rRMSE_h$, see Eq. A2). Relative bias was calculated as in Eq. A3. Performance across all outcome generators was aggregated with mean $rRMSE$ (see Eq. A4) and mean absolute relative bias ($\overline{arb}$, see Eq. A5).

To calculate these metrics, true and estimated generalization performance were calculated for each of the $H$ outcomes with all sets of simulation parameters. For an outcome $h$, $T = 30$ samples of 500 data points were taken of the 3000 simulated data points. Ensemble survival models were trained on the sampled data using the 20 features selected during outcome generation (see Appendix A.1 for more details).

True generalization error for sample $t$ ($Err_{th}$) was obtained by evaluating the resulting models on the remaining 2500 data points with three different performance metrics: $MSE$, $MAE$, $C$-index. Estimates of generalization error ($\tilde{Err}_{th}$) were obtained by applying the four approaches described in Sect. 3.2 to the sample of 500 data points: HO, RRHO, CV, and RRCV.

Using $Err_{th}$ and $\tilde{Err}_{th}$, relative root-mean-squared error ($rRMSE_h$, Eq. A2) and relative bias ($rb_h$, Eq. A3) of the estimation approaches are calculated for each outcome $h$.

$$
rRMSE_h = \sqrt{\frac{1}{T} \sum_{t=1}^{T} \left( \frac{Err_{th} - \tilde{Err}_{th}}{Err_{th}} \right)^2}.
$$ (A2)

$$
rb_h = \frac{1}{T} \sum_{t=1}^{T} \frac{\tilde{Err}_{th} - Err_{th}}{\tilde{Err}_{th} + Err_{th}}.
$$ (A3)

The denominator of Eq. A2 is a departure from typical RMSE calculation, serving to prevent differences in outcome scale or distribution from affecting aggregation of relative error and bias across simulated outcomes.

$$
rRMSE = \frac{1}{H} \sum_{h=1}^{H} rRMSE_h.
$$ (A4)

$$
\overline{arb} = \frac{1}{H} \sum_{h=1}^{H} | rb_h |.
$$ (A5)

These aggregated metrics were used to compare the standard generalization estimation approaches to their RRS counterparts using Welch’s two-sample $t$-test.

A.3 Raw Monte Carlo simulation results

The average error and bias data presented in Fig. 3 are also reported in Tables 1 and 2, respectively.

References

Ardabili, S., Mosavi, A., Várkonyi-Kóczy, A.R.: Advances in machine learning modeling reviewing hybrid and ensemble methods. In: Engineering for Sustainable Future. Springer, Cham, pp. 215–227 (2020). https://doi.org/10.1007/978-3-030-36841-8_21

Austin, P.C.: Generating survival times to simulate cox proportional hazards models with time-varying covariates. Stat. Med. 31(29), 3946–3958 (2012). https://doi.org/10.1002/sim.5452

Bender, R., Augustin, T., Blettner, M.: Generating survival times to simulate cox proportional hazards models. Stat. Med. 24(11), 1713–1723 (2005). https://doi.org/10.1002/sim.2059

Borra, S., Di Ciaccio, A.: Measuring the prediction error. A comparison of cross-validation, bootstrap and covariance penalty methods. Comput. Stat. Data Anal. 54(12), 2976–2989 (2010). https://doi.org/10.1016/j.csda.2010.03.004

Hothorn, T., Buehlmann, P., Dudoit, S., et al.: Survival ensembles. Bioinformatics 7(3), 355–373 (2006)

Hothorn, T., Buehlmann, P., Kneib, T., et al.: mboost: model-based boosting (2022). https://CRAN.R-project.org/package=mboost, R package version 2.9-7

Ishwaran, H., Kogalur, U.: Fast unified random forests for survival, regression, and classification (RF-SRC) (2022). https://cran.r-project.org/package=randomForestSRC, r package version 3.1.1

Kuhn, M., Johnson, K.: Feature Engineering and Selection: A Practical Approach for Predictive Models. CRC Press, Boca Raton (2020)

Rendlemen, M.C., Buatti, J.M., Braun, T.A., et al.: Machine learning with the TCGA-HNSC dataset: improving usability by addressing inconsistency, sparsity, and high-dimensionality. BMC Bioinform. (2019). https://doi.org/10.1186/s12859-019-2929-8

Shi, M., Wang, J., Zhang, C.: Integration of cancer genomics data for tree-based dimensionality reduction and cancer outcome prediction. Mol. Inform. 39(3), 1900–028 (2019). https://doi.org/10.1002/minf.201900028

Smith, B.J.: MachineShop: machine learning models and tools (2021). https://cran.r-project.org/package=MachineShop, r package version 3.1.0

Strobl, C., Boulesteix, A.L., Zeileis, A., et al.: Bias in random forest variable importance measures: illustrations, sources and a solution. BMC Bioinform. (2007). https://doi.org/10.1186/1471-2105-8-25

Strobl, C., Boulesteix, A.L., Kneib, T., et al.: Conditional variable importance for random forests. BMC Bioinform. (2008). https://doi.org/10.1186/1471-2105-9-307

The Cancer Genome Atlas Network: Comprehensive genomic characterization of head and neck squamous cell carcinomas. Nature 517(7536), 576–582 (2015). https://doi.org/10.1038/nature14129

Publisher’s Note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.