RESEARCH

Prediction approaches for partly missing multi-omics covariate data: An empirical comparison study

Jane E. Doe[1*] and John R.S. Smith[1,2]

[*]Correspondence:
jane.e.doe@cambridge.co.uk
[1]Department of Science, University of Cambridge, London, UK
Full list of author information is available at the end of the article

Abstract

Background: Given the increasing availability of omics data, in the last few years more and more *multi-omics* data have been generated, that is, high-dimensional molecular data of several types such as genomic, transcriptomic or proteomic data measured for the same patients. Such data lend themselves to being used as covariates in automatic outcome prediction, because each omics type may contribute unique information, possibly improving predictions compared to only using single omics data types. Frequently, however, the same omics data types are not available for all patients in the training data and in the data to which automatic prediction rules should be applied, the test data. We refer to this type of data as block-wise missing multi-omics data. Currently, the literature on prediction methods applicable to such data is sparse.

Results: Using a collection of 13 publicly available multi-omics data sets, we compare the predictive performances of several recently introduced approaches and some self-developed simple approaches for obtaining predictions in the presence of block-wise missingness in training and test data.

Conclusions: a brief summary and potential implications.

Keywords: multi-omics data; prediction; missing data

Background

The generation of various types of omics data is becoming increasingly fast and cost-effective. As a consequence, there are more and more so-called multi-omics data becoming available, that is, high-dimensional molecular data of several types such as genomic, transcriptomic or proteomic data measured for the same patients. In the last few years several approaches to use these data for patient outcome prediction have been developed (see [1] for an extensive literature review). Nevertheless, doubt has recently been cast over whether there is benefit to using multi-omics data over simple clinical models [2].

Regardless of the usefulness of multi-omics data for prediction, a practical problem is that, for various reasons, multi-omics data from different sources being used for the same prediction problem often do not feature the exact same types of omics data. Most importantly, the test data, that is, the data for which predictions should be obtained, often do not feature the same omics data types as the training data, that is, the data available for obtaining the prediction rule. Another frequent situation is that different subsets of the training data set originating from different sources

feature different omics data types. When focusing on the collection of all omics data types available in at least one of the samples and considering data types not available for the different samples as missing, we can concatenate the data associated with all samples to obtain a large data set with partly missing data. In the following, data concatenated in this form will be referred to as block-wise missing multi-omics data, where the different omics data types will be denoted as blocks. The groups of observations in the data set that share the same combinations of observed data types will be denoted as subsets for simplicity. Note that in addition to the omics data, in practice there are most often clinical features (e.g., age or disease stage) available, which usually contain a lot of predictive information. In this paper we assume that there are always clinical features available, which will be referred to as the clinical block.

The purpose of this paper is to compare different prediction approaches applicable to block-wise missing multi-omics training data and test data in terms of their performance. Currently, there are few approaches available that can be applied to such data, which is why the methods compared in this paper do not only involve existing approaches, but also some self-developed simple approaches. The comparison was performed empirical, based on a large collection of publicly available real multi-omics data sets, for which missing data was generated artificially. As a response variable we used the presence versus absence of a TP53 mutation. While it is not meaningful to predict the presence of TP53 mutations, the latter have been found to poor clinical outcomes [3]. Against this background, we use TP53 as a surrogate for a phenotypic outcome.

The rest of the paper is structured as follows. ... 'Results' section.

Results

Discussion

Conclusions

UEBERNOMMEN AUS BROADER IMPACT SECTION AUS DEM ENTWURF, NOCH ZU BEARBEITEN We believe that our study will help applied researchers confronted with block-wise missing multi-omics data to select suitable methods for obtaining strong predictions models. Moreover, given that methodological literature on prediction methods suitable for such data is still sparse, the results from this study can aid methodological researchers in developing new, stronger methods sharing the strengths of the most promising, while addressing their weaknesses.

Methods

Compared approaches

In this section all approaches considered in the comparison study will be described. In addition to describing the general ideas behind these approaches, some details of their specific configurations in the comparison study presented in this paper will also be given. The following approaches were considered in the study: Complete Case Approach, Single Block Approach, Imputation Approach, MAGIC-LASSO, Block-wise Random Forest, Multi-block Data-Driven sparse PLS, and priority-LASSO-impute. The first three of these are simple approaches, which all use standard random forests [4]. These should serve as a baseline against which the last four methods that are

specially conceptualized to be able to deal with (block-wise missing) multi-omics data are compared. In the following, instead of "block-wise missing multi-omics training data" and "block-wise missing multi-omics test data" the terms "training data" and "test data" will be used for simplicity.

Complete Case Approach A common, simple but ineffective method of dealing with missing data is to remove all samples that contain missing values. However, in the presence of block-wise missingness this would frequently be impossible or very ineffective because there may be none or very few samples with all blocks observed. In order to increase the number of training samples with no missing data, with the Complete Case Approach all those blocks from the training data that do not occur in the test data are removed and then all samples with missing values from the training data are removed. Afterward, a random forest prediction rule is trained on the processed training data. Note that in this approach, as in the majority of the approaches described below, the prediction rule can only be trained after knowing the missing data structure of the test data.

Single Block Approach Even though the available data set may involve several blocks, we may use individual blocks only for training. This may be advantageous in situations in which a single block carries most of the available predictive information or in which the predictive information contained in the different blocks is redundant. In the first step of the single block approach, all blocks not featured in the test data are removed from the training data. Subsequently, a random forest is trained on each block and, using the out-of-bag predictions [4] and the true values of the response variable, performance measure values of these random forests are calculated. The area under the receiver operating characteristic (AUC) is used as a performance measure. Finally, a random forest is trained on the block associated with the largest calculated AUC value.

Imputation Approach A more sophisticated approach to dealing with missing data than the Complete Case Approach is imputing the missing values using a data-driven procedure. An important advantage of the imputation approach over the complete cases approach is that no samples have to be excluded when training the prediction rule. The popular missForest algorithm [5] is used for imputation, as this method can deal with both numeric and categorical features and has been seen to outperform competitors in a comparison study presented in the same paper. First, the whole training data set is imputed, second, all blocks in the training data set not available in the test data set are removed and, third, a random forest is constructed using the training data set.

Missingness Adapted Group Informed Clustered (MAGIC)-LASSO The MAGIC-LASSO [6] is an iterative fitting procedure based on the Group-LASSO. As a first step, features with large proportions of missing values and unnecessary information are removed from the training data. Subsequently, those features are removed for which the missingness proportions differ strongly between training and test data. Next a clustering algorithm is applied to cluster the features into groups with similar missingness patterns in an effort to maximize the numbers of complete cases

in each group. Group-LASSO is then applied separately to each clustered group to select features that likely are informative. The clustering is subsequently applied to the reduced set of features and again, separate Group-LASSO models are fitted to each resulting group in order to further reduce the feature space. This process of clustering and feature selection using Group-LASSO models is repeated iteratively until a parsimonious set of features has been obtained. These features are then sorted according to their proportions of missing values, starting with the feature with least missing values. Then a sequence of Group-LASSO models is fitted to these features, starting with only the first feature, then the first two features, the first three features, and so on. A cross-validated performance estimate is obtained for each model. The procedure is stopped as soon as there are no complete cases anymore. As a last step the model with the best cross-validated performance estimate is selected as the final model.

Block-wise Random Forest Unlike the approaches described above, the method introduced by [7], denoted "Block-wise Random Forest" in the following, uses all available measurements from the training data set without performing imputation. A separate random forest is grown for each block, where in each case all samples that feature measurements for the corresponding block in the training data are used. In order to obtain predictions for the test data, first the corresponding random forests are applied to each block available in the test data and in each case a predicted probability for $Y = 2$ is calculated, where $Y \in \{1, 2\}$ describes the binary response. Second, a weighted average of the predicted probabilities obtained using each test data block is calculated with weights proportional to the out-of-bag calculated AUC values of the corresponding random forests. Note that, in general, other weights can be used as well, for example weights based on performance measures other than the AUC or equal weights for all blocks, where the latter would correspond to an unweighted average.

Multi-block Data-Driven sparse PLS (mdd-sPLS) Like the imputation approach described above, mdd-sPLS, introduced by [8], involves imputing the data set, but unlike the imputation approach, not only the training data set is imputed, but also the test data set. Therefore, all blocks can be used for prediction. The mdd-sPLS algorithm is a complex partial least squares (PLS) based method. Put simply, mdd-sPLS performs PLS type procedures separately on the blocks, combines the information afterward and predicts the outcome. The missing values in training and test data of those features that are used by mdd-sPLS in prediction are imputed using additional PLS based models. The missing values are imputed using the outcome and the other covariates, respectively. These two steps, building the prediction model and imputing missing values, are repeated until convergence of the involved latent variables.

priority-LASSO-impute (pL-imp (available)) The pL-imp (available) algorithm, developed by [9], which will soon be available in the R package 'priorityLASSO' [10], is an extension of the multi-omics prediction method priority-LASSO [11] for block-wise missing multi-omics data. Priority-LASSO is a method based on the

Least Absolute Shrinkage and Selection Operator (LASSO) that allows researchers to specify a priority ranking of the blocks. Such a priority ranking is often given, because some blocks are more established or easier to obtain than others. The first step of obtaining the priority-LASSO prediction rule is to fit a LASSO model using only the features in the block with highest priority. In the second step, again a LASSO model is fit, however, this time using only the features in the block with second-highest priority using the linear predictor from the LASSO model fitted in the first step as an offset in the model equation. By including the latter offset, only that part of the predictive information contained in the block with second-highest priority that is not contained in the block with highest priority is used. This process is continued iteratively for all blocks in the order of their priority, this way obtaining estimated model coefficients for all features.

In the case of block-wise missing multi-omics data, this estimation scheme would not be applicable, because in each step the offsets would not be available for all observations. [9] considered several solutions to this issue. In this paper only one of these will be used, pL-imp (available), because it showed good results in [9] and also seems to be the most promising from a conceptual point of view. With this approach, the missing offsets are imputed using linear regression. In simple terms, first, in each step this approach learns a linear regression model using observations for which the offsets are available, where the offsets are the response variables and other blocks contain the covariates. Second, these linear regression models are used to predict the missing offsets. More precisely, a LASSO model is trained that uses that combination of blocks as covariates that maximises the number of observations available for fitting the model.

The pL-imp (available) algorithm was not designed to handle missing blocks in the test data. In order to deal with this issue, all blocks that are not available in the test data are excluded before fitting the model to the training data. Moreover, the pL-imp (available) algorithm, as the original priority-LASSO algorithm, requires the user to provide a priority ranking of the available blocks. As no useful biological information is available for the data sets considered in the comparison study performed for this paper, the priority rankings will be determined in the following way: 1) Fit a LASSO model to each block and estimate the AUC value associated with each model using 5-fold cross-validation (CV); 2) Assign the highest priority to the block associated with the largest cross-validated AUC value, the second highest priority to the block associated with the second-largest cross-validated AUC value and so on; in cases in which two or more blocks are associated with the same cross-validated AUC value, the priority order between these blocks is assigned randomly.

Data

The data material consists of 13 publicly available multi-omics data sets from The Cancer Genome Atlas (TCGA) project. These data are a subset of 21 data sets previously used in [1]. From these 21 data sets 18 contained all four blocks that were considered as covariates (see below) and the mutation block that contained the information on the response variable, that is, the presence versus absence of the TP53 mutation. From the remaining 18 data sets we removed imbalanced data sets for which the smaller response variable class was represented by less than 15% of

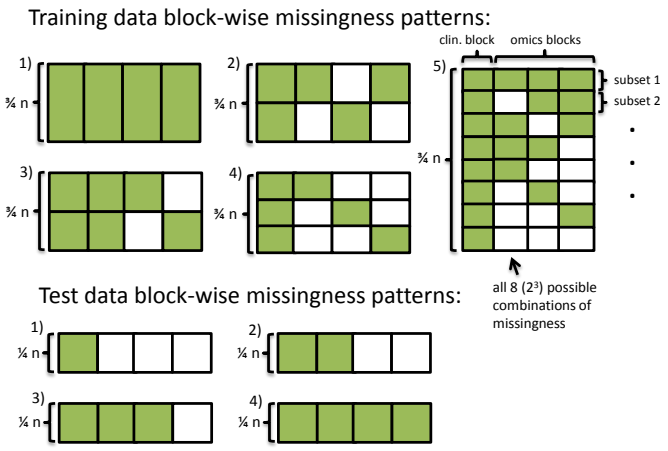

Figure 1 Block-wise missingness patterns in training and test data. Colored/empty boxes indicate that the respective blocks are present/absent. The first column always represent the clinical block.

the observations. This resulted in 13 data sets to be used in the comparison study. The pre-processed versions of the data sets available in the electronic appendix of [1] were used, for details on the pre-processing see the paper. Each of the data sets featured the same five blocks: clinical block, copy number variation block, miRNA block, mutation block, and RNA block. The mutation data were excluded because it is not meaningful to predict the TP53 mutation using other mutations. Table ??? gives an overview on the used data sets.

Design of the comparison study

As mentioned above, block-wise missingness patterns are generated by randomly deleting parts of the data sets. In order to avoid overoptimistic performance estimates, the data sets are split repeatedly into training and test data in the ratio 3:1. The block-wise missingness patterns are induced separately in training and test data, where there are five different patterns for the training data sets and four for the test data sets, see Figure 1. As is evident from Figure 1, each of the missingness patterns in the training data sets consists of either one, two, three or eight subsets of samples. For each division into training and test data, the subset memberships of the training samples are assigned randomly and the subsets are of equal size for each training data set. Moreover, to avoid a dependence of the results on the succession of the blocks in the missingness patterns in Figure 1, both for the training and the test data, the identities of the blocks are assigned at random for each division into training and test data. For example, consider block-wise missingness pattern 2) for the training data and block-wise missingness pattern 3) for the test data: Here, for the first division into training and test data, the first set in the training data might include RNA and miRNA data and the second set only mutation data, whereas the test data might include RNA and mutation, but no miRNA data. But for the second division, the first set in the training data might include only mutation and miRNA data and the second set only RNA data, while the test data may include mutation and miRNA data, but no RNA data.

For each combination of training data and test data missingness patterns, five divisions in training and test data of each data set are generated randomly inducing the respective missingness patterns. For each division, the methods described in 'Compared approaches' section are learned on the training data, subsequently applied to the test data, and the AUC value between the class probability predictions and the actual class memberships is calculated. Subsequently, these AUC values are averaged over the five repetitions and again averaged over the 13 data sets. Thus, a single AUC value is obtained for each method and each combination of training and test data set missingness pattern, that is, there are $7 \times 5 \times 4 = 140$ AUC values in total. Note that, if there are no missing blocks in the training data, that is, for training data block-wise missingness pattern 1), the Imputation Approach and the Complete Case Approach are identical. The same is true for priority-LASSO-impute and standard priority-LASSO.

Details on the configurations used for the compared methods

For the random forests used in the Complete Case Approach, the Single Block Approach, the Imputation Approach, and the Block-wise Random Forest no tuning parameter optimization was performed, but the tuning parameter values were set to the default values of ranger R package (version ????). For example, the numbers of features sampled for each split *mtry* were set to the square roots of the numbers of features. Given the high computational cost of performing missForest, only one iteration was performed for this algorithm in the Imputation Approach. Moreover, for the same reason, a relatively small number of 25 trees was used in this algorithm. The shrinkage parameters in the LASSO models involved in the priority-LASSO and priority-LASSO-impute estimation procedures were determined using grid search and 10-fold CV.

Appendix

Text for this section...

Acknowledgements
Text for this section...

Funding
Text for this section...

Abbreviations
Text for this section...

Availability of data and materials
Text for this section...

Ethics approval and consent to participate
Text for this section...

Competing interests
The authors declare that they have no competing interests.

Consent for publication
Text for this section...

Authors' contributions
Text for this section ...

Authors' information
Text for this section...

Author details
[1]Department of Science, University of Cambridge, London, UK. [2]Institute of Biology, National University of Sciences, Kiel, Germany.

References

1. Hornung R, Wright MN. Block Forests: random forests for blocks of clinical and omics covariate data. BMC Bioinformatics. 2019;20:358.
2. Herrmann M, Probst P, Hornung R, Jurinovic V, Boulesteix AL. Large-scale benchmark study of survival prediction methods using multi-omics data. Briefings in Bioinformatics. 2020. Bbaa167.
3. Wang X, Sun Q. TP53 mutations, expression and interaction networks in human cancers. Oncotarget. 2017;8(1):624–643.
4. Breiman L. Random Forests. Mach Learn. 2001;45(1):5–32.
5. Stekhoven DJ, Bühlmann P. MissForest—non-parametric missing value imputation for mixed-type data. Bioinformatics. 2011;28(1):112–118.
6. Gentry AE, Kirkpatrick RM, Peterson RE, Webb BT. Missingness Adapted Group Informed Clustered (MAGIC)-LASSO: A novel paradigm for prediction in data with widespread non-random missingness. bioRxiv. 2021. Available from: `https://www.biorxiv.org/content/early/2021/04/30/2021.04.29.442057`.
7. Krautenbacher N. Learning on complex, biased, and big data: disease risk prediction in epidemiological studies and genomic medicine on the example of childhood asthma [Dissertation]. Technical University of Munich; 2018. Available from: `http://mediatum.ub.tum.de/doc/1446834/document.pdf`.
8. Lorenzo H, Saracco J, Thiébaut R. Supervised Learning for Multi-Block Incomplete Data; 2019. arXiv:1901.04380.
9. Hagenberg J. Penalized regression approaches for prognostic modelling using multi-omics data with block-wise missing values [Master's Thesis]; 2020.
10. Klau S, Hornung R, Bauer A. prioritylasso: Analyzing Multiple Omics Data with an Offset Approach; 2020. R package version 0.2.4. Available from: `https://github.com/RomanHornung/prioritylasso`.
11. Klau S, Jurinovic V, Hornung R, Herold T, Boulesteix AL. Priority-Lasso: a simple hierarchical approach to the prediction of clinical outcome using multi-omics data. BMC Bioinformatics. 2018;19:322.

Figures

Figure 2 Sample figure title

Figure 3 Sample figure title

Tables

Table 1 Sample table title. This is where the description of the table should go

	B1	B2	B3
A1	0.1	0.2	0.3
A2	...	..	.
A3	..	.	.

Additional Files

Additional file 1 — Sample additional file title
Additional file descriptions text (including details of how to view the file, if it is in a non-standard format or the file extension). This might refer to a multi-page table or a figure.

Additional file 2 — Sample additional file title
Additional file descriptions text.