Ludwig-Maximilians-University
Munich

Institute for Statistics

Master Thesis

A comparison study of prediction approaches for multiple training data sets and test data with block-wise missing values

Author:
Frederik
Ludwigs

Supervisor:
Dr. Roman
Hornung

March 15, 2020

Abstract

Contents

List of Figures

List of Tables

1 Introduction

On October 1, 1990 the international scientific research project named *Human Genome Project* was launched, with the aim to sequence the first complete human genome ever [1]. After investments of totally $2.7 billion and 13 years of research the sequencing was officially finished in 2003 [2]. Since then, on the one hand, there have been biomedical advances that have led to the identification of disease genes, "leading to improved diagnosis and novel approaches in therapy" [[3], p. 14]. And on the other hand there has been an "extraordinary progress [...] in genome sequencing technologies" [[4], p. 333] leading to a sharp drop in sequencing prices. Nowadays whole genome sequencing is available and affordable for almost everyone - e.g. 'Veritas Genomics', offers whole genome sequencing for ∼$700 [5].

Besides the 'genome' that carries the whole genetic material of an organism, there are also other types of '-omes', such as 'epigenomes', 'transcriptomes', 'proteomes' and 'microbiomes'. The time and costs to collect data from these different types of '-omes' has been reduced drastically ever since the completion of the Human Genome Project [[6], [7], [8], [9], [10], [11]]. The methods for "fast, automated analyses of large numbers of substances including DNA, RNA, proteins, and other types of molecules" [12] are summarized under the term 'High Throughput Technologies'. These technologies have made data from molecular processes available for many patients on a large scale.

The collection of any type of '-omes' data is commonly called 'omics data'. In the clinical context it is highly interesting to incorporate such omics data into different statistical approaches. A common example in this context is the survival time prediction for cancer patients, where additionally to regular clinical data (e.g. 'weight', 'height', 'blood pressure') gene expression data has been incorporated into the survival models. This additional omics data has "often been found to be useful for predicting survival response[s]" [[7], p. 1]. In "the beginning, only data from single omics was used to build such prediction models, together or without [...] clinical data" [[13], p. 1]. The usage of multiple different types of '-omes' in a single prediction approach was the next logical step and coined the term 'multi-omics data'. The theoretical aspects of integrating several omics types into a single prediction model and how to deal with the block-wise structures has been topic of several papers already - e.g. [13], [14], [15], [16], [17].

This thesis deals with a special type of missing data "that is common in practice, particular in the context of multi-omics data" [18], the *block-wise missingness*. Data with block-wise missingness consists of different folds and feature-blocks. While a feature-block stands for a collection of associated covariates, a fold represents a set of observations with the same observed feature-blocks. In data sets with block-wise missingness there is always at least one fold that misses at least one of the feature-blocks, so that not all observations have the same observed blocks. It can for example arise when concatenating multiple training sets, whereby these have the same target variable, but still different feature-blocks [18].

Most statistical methods require fully observed data for their training and predictions. In data with block-wise missingness this requirement is clearly not met, so that either the approaches need methodical adjustment or the data needs to be processed, so that a model can be trained on it regularly. This emerges the following challenges: How can we fit a model on the data, without removing observations or whole feature-blocks? How does a model that uses complete cases only perform in comparison? Does imputation work properly in these settings? How does a model that uses single feature-blocks only perform in comparison? How can a model do predictions on observations with missing feature-blocks?

Additional to the problem of block-wise missingness, there is also the challenge of “inherent high dimensionality” [[19] p. 93], when working with multi-omics data. Data from a single omics type can easily exceed 10,000 covariates and the corresponding data sets usually consist of less observations than features [13]. Besides the predictive performance of an approach it is furthermore important for the approach to be sparse. “Sparsity is [...] an important aspect of the model which contributes to its practical utility” [[15], p. 3], as it makes the model much more interpretable than models including several thousands of variables.

A method that handles high dimensional data, even if the number of observations is lower than the amount of features, is the random forest [13]. It can also handle different input types, does not need a lot of tuning and yields comparable predictive performances [20]. The only drawback of this method is that it is not as interpretable as “models yielding [*in*] coefficient estimates of few relevant features” [[13], p. 35], as penalised regression approaches for example. Nevertheless variable importance measures can be extracted with the random forest method, aswell as partial dependencies. Furthermore the random forest method has been used successfully in various articles dealing with multi-omics data - e.g. [13], [14]. Additionally there have been proposals by R. Hornung et al. [18] and N. Krautenbacher [19] that modify the random forest approach, so that it can directly deal with block-wise missing data.

The different adaptions of penalised regression, as for example the IPF-Priority-Lasso [7] [15] can be modified so that they can handle block-wise missingness. The theoretical aspects of these approaches are not part of this thesis, but of J. Hagenberg’s [21]. Nevertheless the performances of the random forest adaptions and the penalised regression adaptions are compared in this thesis aswell.

Even tough the problem of block-wise missingness is common in multi-omics data there are, to our knowledge, no comparison studies of such prediction approaches yet. N. Krautenbacher has already stated that “reliable analysis strategies for multi-omics data [...] [with block-wise missingness are] urgently needed” [[19] p. 94]. The thesis at hand aims to provide such a large scale comparison study of prediction approaches capable to deal with block-wise missingness and shall help to find reliable strategies.
For this the predictive performance of two naive random forest approaches, a random forest based imputation approach, two random forest adaptions and the adaptions of penalised regression are compared. In the 'Methods' chapter, firstly the term 'block-wise missingness' is defined in more detail and how it can arise in multi-omics data. Then a brief theoretical explanation of the random forest method for classification is given. Following three approaches are explained that process the data, so that a regular random forest can be trained with it. Moreover two methodological adaptations of the random forest method are illustrated. These methodological adaptations let the random forest approach directly deal with block-wise missingness and do not need any processing of the data. In the section 'Benchmark Experiment' the three different data sources used to measure the performances of the various approaches are described and investigated. The metrics and tactic used for the evaluation of the models is given afterwards. In the 'Results' chapter all approaches are analysed and compared. In last section all findings of the thesis are discussed, conclusions are drawn and an outlook is given.

2 Methods

This section deals with the theory of the random forest approach and how it can be adapted to handle data with block-wise missingness.
In the beginning, it is described, what block-wise missingness is and how it can arise in multi-omics data. Then the theory of the random forest method for classification is illustrated. Subsequent three approaches that process the data, so that a regular random forest can be fit on them, are described. In the end, two adaptions of the random forest method that enable them to directly deal with block-wise missingness, are shown.

2.1 Block-wise missingness in multi-omics data

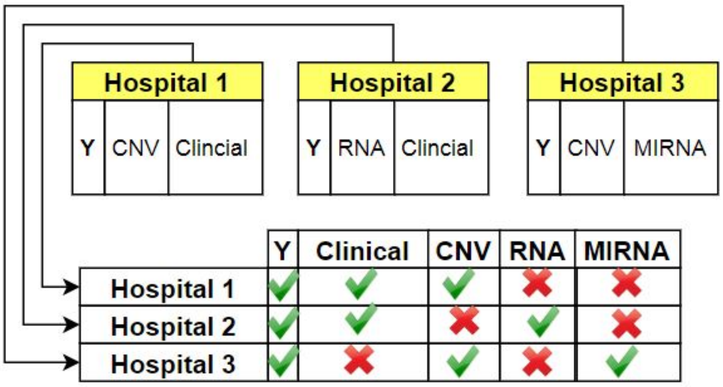
Collecting omics data has become significantly cheaper and faster ever since the completion of the Human Genome Project. As a result, these data types are used more and more frequently in the biomedical research - e.g. risk prediction for childhood asthma [19]. Even though the integration of multiple omics types into a single prediction approach seems promising there are still challenges to face. One is a special type of missingness that is common in the context of multi-omics data, the so called block-wise missingness [18].
Before clarifying how block-wise missingness can arise in multi-omics data and the challenges that come with it, lets have a closer look at block-wise missingness in general. To make explanations more descriptive, table 1 shows an example for a data set with block-wise missingness. The example data consists of eight observations and 105 covariates in total. While the covariates 'weight', 'height', 'income' and 'education' are pretty much self-explanatory, the features 'g_1', ..., 'g_{100}' could for example represent measurements from the epigenome. Data sets with block-wise missingness always consist of different *blocks* and *folds*. On the on hand, a **block** describes a set of covariates containing all features collected on the basis of a characteristic - basically all covariates that are related in content. The example data in table 1 has three blocks in total. 'Block 1' consists of the variables 'weight' and 'height' representing the physical properties. 'Block 2' contains the variables 'income' and 'education' standing for economic properties. 'Block 3' includes the remaining variables 'g_1', ..., 'g_{100}' and represents biological properties. On the other hand, a **fold** represents a set of observations with the same observed feature-blocks - basically all observations with the same observed features. The data set in table 1 consists of three folds in total. 'Fold 1' holds the observations 1, 2 and 3, as these have the same observed feature-blocks ('Block 1' & 'Block 2'). 'Fold 2' holds observations 4 and 5, while 'Fold 3' consists

of the remaining observations 6, 7 and 8. As each fold has different observed blocks, each fold is unique and every observation belongs to exactly one of them. The only variable all folds must have in common is the target variable - 'Y' in the example data set in table 1.

ID	*weight*	*height*	*income*	*education*	g_1	$\cdots$	g_{100}	*Y*	
1	65.4	187	2.536	*Upper*		$\cdots$		1	Fold1
2	83.9	192	1.342	*Lower*		$\cdots$		0	
3	67.4	167	5.332	*Upper*		$\cdots$		1	
4			743	*Lower*	−0.42	$\cdots$	1.43	1	Fold2
5			2.125	*Lower*	0.52	$\cdots$	−1.37	0	
6	105.2	175			−1.53	$\cdots$	201	0	Fold3
7	71.5	173			0.93	$\cdots$	0.53	0	
8	73.0	169			0.31	$\cdots$	−0.07	1	
	*Block*1		*Block*2		*Block*3				

Table 1: A minimalist example for a data set with block-wise missingness

This type of missingness can arise quite quickly, when working with multi-omics data. There are two main reasons for the block-wise missingness in multi-omics data. The first one is related to the costs of collecting omics data. Even though these have been reduced drastically over the last 15 years, collecting omics data is still more complex and expensive then obtaining clinical data for example. As a consequence, due the financial constraints, omics data can not always be collected for all participants of a study. Therefore patients from the same study can end up with different observed blocks, so that the data set for the whole study contains block-wise missingness. The second reason is the collection of omics-data from different sources - e.g. various hospitals. Even though the sources do research regarding the same disease the surveyed omics blocks can still differ. Therefore the concatenation of these results in a data set with block-wise missingness. For a better understanding of this scenario, it is illustrated in figure 1. In the top of the figure are the different data sources *(Hospital 1-3)*. Each of these consists of the target variable 'Y' and two feature-blocks - e.g. 'Hospital 2' consists of 'Y' and the feature-blocks 'RNA' and 'Clinical', whereby 'RNA' represents multidimensional RNA measurements and 'Clinical' several clinical features. Even though the target variable 'Y' is the same for all sources, the collected feature-blocks are still different. The concatenation of such diverse sources results in a data set with block-wise missingness then. The concatenated data consists of three unique folds and four different feature-blocks. In the concatenated data in figure 1 an observed block is marked with a green tick and a missing block with a red cross. The observations from the fold 'Hospital 2' only have 'RNA' and 'Clinical' as observed feature-blocks, so that they miss all the features from the blocks 'CNV' and 'MIRNA' in the concatenated data.



	Y	Clinical	CNV	RNA	MIRNA
Hospital 1	✓	✓	✓	✗	✗
Hospital 2	✓	✓	✗	✓	✗
Hospital 3	✓	✗	✓	✗	✓

Figure 1: Example for block-wise missingness, when concatenating data from diverse sources

It should be obvious that training a prediction model, as for example a random forest, is not directly possible on data with block-wise missingness. Either the methods have to be adopted or the data processed - e.g. removing folds. The predictions also have their challenges, as block-wise missingness can affect the test data aswell. This raises the question, how a model can do predictions for observations that miss blocks the model has been trained with. These challenges have to be taken into account when proposing methods capable to deal with block-wise missingness.
The remaining chapters focus on approaches and adaptions to make model fitting on such data possible. Firstly the concept of the random forest for classification is explained and then the methodical adaptions based on it.

2.2 Random Forest

This chapter illustrates the random forest method that has already been applied in several articles dealing with multi-omics data [[13], [14], [15]]. It is a "powerful prediction method [...] able to capture complex dependency patterns between the outcome and the covariates" [[14] p. 2]. Furthermore it does not need a lot tuning, deals with different types of data and naturally handles high-dimensional data, with more covariates than observations

[13]. Additionally the random forest method can be applied to classification-, regression- and even to survival-problems. The latter was added to the random forest method in 2008 by Ishwaran et al. [22]. As this thesis focuses on classification tasks, only the random forest for classification is explained.
The random forest is an tree-based ensemble method that was introduced by L. Breimann in 2001 [23]. An ensemble is a concept from machine learning that "train[s] multiple models using the same learning algorithm" [24]. Therefore an ensemble consists of η identical learners - in case of the random forest these learners are decision trees. To meaningfully train η equal learners, a different training set is needed for each of them, else the learners would be completely identical. To generate η different training sets bagging - standing for 'Bootstrap Aggregation' - is applied to the original data. This samples n observations with replacement from the original data set to create an own data set for each of the η learners [24]. On each of the boostrapped data sets a learner can be trained then. To create predictions with such an ensemble, the predictions from each of the η learners are averaged. The advantage of an ensemble is that it "can dramatically reduce the variance of unstable procedures [...] leading to improved prediction[s]"[[20] p. 283].
A decision tree is an excellent base learner for an ensemble, as it can capture complex interactions and have a relative low bias, if grown sufficiently deep. Especially as single trees are known to be noisy, they benefit from the ensemble [20]. So decision trees are the basis of random forest method and therefore it is crucial to understand how these work in order to properly understand the random forest method.

2.2.1 Decision Trees

A decision tree is a supervised learning method that applies recursive binary splitting. This binary splitting "partition[s] the feature space into a set of rectangles" [[20] p. 305], such that each rectangle is as pure as possible in terms of the response. To receive a prediction for a test observation you have to assign the test observation - based on its features - to one of the rectangles in the partitioned feature space. The prediction then equals the response's distribution of the training observations within this rectangle.
To make the algorithm easier to understand the single partition steps for a classification tree are shown in figure 2. The figure totally consists of three plots. They all show the scatter-plot of 'weight' and 'height' from table 1 for the observations from 'Fold 1' and 'Fold 3'. The observations with a positive outcome are marked as blue. In each plot the segments are annotated

with the Node that contains the observations. In the first plot of figure 2 all observations are within one segment and therefore in one node - 'Node 1'. Within this segment there are three observations with a positive and three with a negative response - hence the class distribution is 50/50. To meaningfully split the data into two sub-regions the algorithm tries every possible split value for each feature. For each of these possible splits, it can be measured how pure the resulting sub-regions are - in terms of the response. If no stopping criterion is fulfilled, then the data is split into the most pure sub-regions. In the example in figure 2 the fist split variable is chosen as 'weight' with the value 69. Therefore the data from 'Node 1' is split into 'Node 2' and 'Node 3'. 'Node 2' only contains observations with a weight ≥ 69 and 'Node 3' only observations with a weight < 69. 'Node 2' has a class distribution of 25/75 and 'Node 3' is completely plain, as it only contains observations with a positive outcome - 100/0. So both resulting nodes are more pure than the the original node. As 'Node 3' is completely pure it can not further be split. 'Node 2' is not pure, so that the algorithm tries all possible splits on this feature space to make it more pure - it selected the variable 'weight' with the value 171. 'Node 2' is therefore further split into 'Node 4' and 'Node 5'. Both resulting nodes are completely pure and can not be further split. In summary, the decision tree algorithm tries to split the feature space, such that the resulting sub-spaces are as plain as possible regarding the response. This is done with an exhaustive search, trying all possible split values and choosing the split that maximises the purity of the resulting nodes.

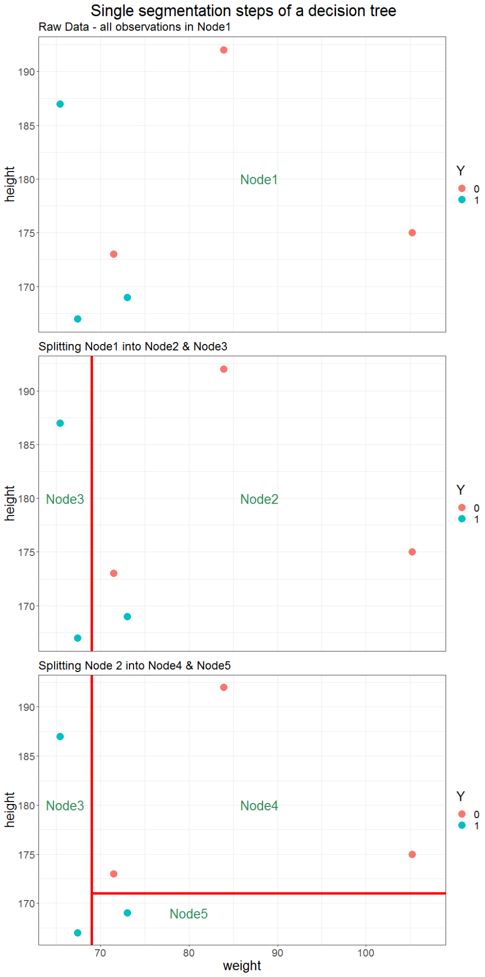


Figure 2: Example for the splitting of a two-dimensional feature space by a decision tree

The same information as in figure 2 can be displayed much easier - especially for high dimensionalities - as a decision tree. The corresponding decision tree for figure 2 is displayed in figure 3. Each node displays the amount observations it contains and the response’s distribution of these. The first node for example contains 100% of the observations, and has 50% positive and 50% negative responses. The data from this node is split by the ’weight’ variable and results in the child nodes ’Node 2’ (observations with weight $\geq$ 69) and ’Node 3’ (observations with weight $<$ 69). As ’Node 2’ is further split into ’Node 4’ and ’Node 5’ it is called internal node. It is split by the height variable and results in ’Node 4’ (height $\geq$ 171) and ’Node 5’ (height $<$ 171). As the nodes 3, 4 and 5 can not be further split, these are called ’Terminal Nodes’.

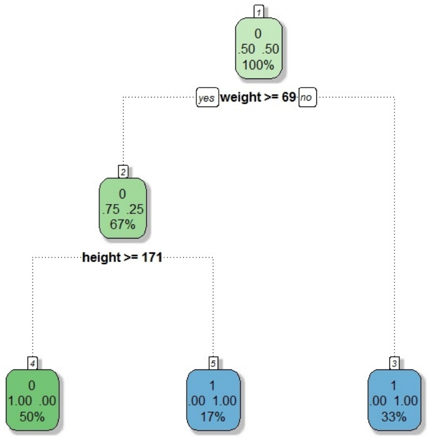


Figure 3: Corresponding decision tree for figure 2

2.3 Reference Approach 1 - single block

2.4 Reference Approach 2 - complete case analysis

2.5 Imputation Approach - miss Forest

2.6 Adjustment Approach 1 - block-wise RF

2.7 Adjustment Approach 2 - fold-wise RF

3 Benchmark Experiments

3.1 Datasets

3.1.1 Own data

Subsetting the omics blocks

Inducing blockwise missingness

3.1.2 Data from Hagenberg's thesis

3.1.3 Real data

3.2 Accessing the performance

3.2.1 CV - Test-Situations

3.2.2 Metrics

4 Results

4.1 Own data

4.1.1 Scenario 1 - names to be adjusted

4.1.2 Scenario 2 - names to be adjusted

4.1.3 Scenario 3 - names to be adjusted

4.1.4 Scenario 4 - names to be adjusted

4.2 Data from Hagenberg's thesis

4.3 Real data

5 Discussion and Conclusion

6 Bibliography

[1] Francis S Collins. “Medical and societal consequences of the Human Genome Project”. In: *New England Journal of Medicine* 341.1 (1999), pp. 28–37.

[2] *National Human Genome Research Institute.* `https://www.genome.gov/about-genomics/fact-sheets/Sequencing-Human-Genome-cost`. Accessed: 2020-01-07.

[3] Belinda JF Rossiter and C Thomas Caskey. “Impact of the Human Genome Project on medical practice”. In: *Annals of surgical oncology* 2.1 (1995), pp. 14–25.

[4] Sara Goodwin, John D McPherson, and W Richard McCombie. “Coming of age: ten years of next-generation sequencing technologies”. In: *Nature Reviews Genetics* 17.6 (2016), p. 333.

[5] *Veritas - The Genome Company.* `https://www.veritasgenetics.com/myGenome`. Accessed: 2020-01-19.

[6] Ke Bi et al. “Transcriptome-based exon capture enables highly cost-effective comparative genomic data collection at moderate evolutionary scales”. In: *BMC genomics* 13.1 (2012), p. 403.

[7] Anne-Laure Boulesteix et al. “IPF-LASSO: Integrative-penalized regression with penalty factors for prediction based on multi-omics data”. In: *Computational and mathematical methods in medicine* 2017 (2017).

[8] Valeria D’Argenio. “The high-throughput analyses era: are we ready for the data struggle?” In: *High-throughput* 7.1 (2018), p. 8.

[9] Gregory B Gloor et al. “Microbiome profiling by illumina sequencing of combinatorial sequence-tagged PCR products”. In: *PloS one* 5.10 (2010).

[10] Shrutii Sarda and Sridhar Hannenhalli. “Next-generation sequencing and epigenomics research: a hammer in search of nails”. In: *Genomics & informatics* 12.1 (2014), p. 2.

[11] Forest M White. “The potential cost of high-throughput proteomics”. In: *Sci. Signal.* 4.160 (2011), pp. 8.

[12] *National Institutes of Health.* `https://commonfund.nih.gov/arra/highthroughput`. Accessed: 2020-01-30.

[13] Moritz Herrmann. “Large-scale benchmark study of prediction methods using multi-omics data”. PhD thesis. 2019.

[14] Roman Hornung and Marvin N Wright. “Block Forests: random forests for blocks of clinical and omics covariate data”. In: *BMC bioinformatics* 20.1 (2019), p. 358.

[15] Simon Klau et al. “Priority-Lasso: a simple hierarchical approach to the prediction of clinical outcome using multi-omics data”. In: *BMC bioinformatics* 19.1 (2018), p. 322.

[16] Stefanie Hieke et al. “Integrating multiple molecular sources into a clinical risk prediction signature by extracting complementary information”. In: *BMC bioinformatics* 17.1 (2016), p. 327.

[17] Qing Zhao et al. “Combining multidimensional genomic measurements for predicting cancer prognosis: observations from TCGA”. In: *Briefings in bioinformatics* 16.2 (2015), pp. 291–303.

[18] Roman Hornung et al. “Random forests for multiple training data sets with varying covariate sets”. manuscript - unpublished yet. - in prep.

[19] Norbert Krautenbacher. “Learning on complex, biased, and big data: disease risk prediction in epidemiological studies and genomic medicine on the example of childhood asthma”. PhD thesis. Technische Universität München, 2018.

[20] Trevor Hastie, Robert Tibshirani, and Jerome Friedman. *The elements of statistical learning: data mining, inference, and prediction.* Springer Science & Business Media, 2009.

[21] Jonas Hagenberg. “Penalized regression approaches for prognostic modelling using multi-omics data with block-wise missing values”. manuscript - unpublished yet. - in prep.

[22] Hemant Ishwaran et al. “Random survival forests”. In: *The annals of applied statistics* 2.3 (2008), pp. 841–860.

[23] Leo Breiman. “Random forests”. In: *Machine learning* 45.1 (2001), pp. 5–32.

[24] *What is the difference between Bagging and Boosting?* `https://quantdare.com/what-is-the-difference-between-bagging-and-boosting/`. Accessed: 2020-03-06.

7 Attachment

Figures

Tables