University of Oxford

MSc in Statistical Science

Final thesis

Missing data imputation for Haemorrhagic shock prediction

Author:
Antoine Ogier

Supervisor:
Pr. Julie Josse
(École polytechnique)
Pr. Geoff Nicholls
(University of Oxford)

September 2018

Abstract

Hemorrhagic shock is a condition that can be life-threatening but that has much higher survival rates if treated early. It is also quite difficult to detect. Because of this, there is a strong case for a tool that predicts it based on prehospital measurement made on trauma patients.

The Traumabase dataset provides a large history of such measurements, and could be used to learn a model to predict hemorrhagic shock (Chapter 1). However, the presence of missing data complicates the task. In this work, we specifically explore one way to handle missing data: imputation of unobserved values (Chapter 2). In a context where the final goal is prediction on new real-world patients, rather than parameter estimation, there are some important differences that we investigate. In particular, current implementations of imputation methods need to be modified to work in such case (Chapter 3). The possible presence of missing values for new patients at the time of prediction (in addition to those in the records) means that some issues beyond parameter estimation appear (Chapters 4 and 5).

After investigating the issues linked to imputation in this context, we go back to the Traumabase and estimate its potential for hemorrhagic shock prediction (Chapter 6).

Acknowledgements

I would first like to thank Pr. Julie Josse and Pr. Geoff Nicholls who supervised my work and were always available to talk about the challenges I encountered.

I would also like to thank Pr. Jean-Pierre Nadal, Dr. Sophie Hamada and Dr. Tobias Gauss for their expertise on the Traumabase and there friendliness throughout the project.

Thank you to Morgane for proofreading my work.

Contents

1 Goal and data **3**
1.1 Hemorrhagic shock: a lethal but preventable condition . . . 3
1.2 The Traumabase data 5
1.3 Objective and formalization 13

2 Imputation **17**
2.1 Missing data mechanisms 17
2.2 Main types of imputation 18
2.3 Multiple imputation 19

3 Methodology: imputation and the validation split **23**
3.1 Empirical risk minimization and cross-validation 23
3.2 ERM with missing data: the problem of current methodologies 25
3.3 Possible solutions 27

4 Impact of missing data: the case of linear regression 33
4.1 Problem set-up 33
4.2 Analysis 36

5 Imputation and prediction: Empirical findings 41
5.1 Impact of missing data 41
5.2 Multiple imputation 44

6 Imputing the Traumabase data for prediction 47
6.1 Methodology 47
6.2 Results 50

Conclusion 53

Appendix A Simulated normal data 55

Appendix B Abalone data 57

Bibliography 59

Chapter 1

Goal and data

1.1 Hemorrhagic shock: a lethal but preventable condition

1.1.1 Description

Post-traumatic bleeding is the primary cause of preventable deaths among injured patients around the world [1][13]. When a person sustains a severe injury (e.g. due to a car accident or violent assault), she may present serious internal or external bleeding. If the injury is serious enough, the natural coagulation process is not sufficient to stop the blood loss. In that case, if too much blood is lost, the patient enters a state called hemorrhagic shock (HS) where the body is no longer able to provide vital organs with enough dioxygen to sustain them [9]. At this point, even proper care may not be enough to save the patient [15].

When an individual sustains an injury, she is usually first taken in charge by first responders who evaluate the situation and the patient's state, before transporting her to a trauma centre when the patient will be treated. To prevent patients from falling into HS, procedures have been established in most hospitals to trigger a fast response to suspected haemorhhage [41]: massive transfusion (MT) protocols can be activated even before the patient enters the hospital [36]. When this happens, the hospital gets ready to transfuse the patient with large amounts of blood as soon as she arrives, and frees up the necessary personnel. This way, the time interval between the injury and the transfusion in minimal. Studies [23] [37] show that early transfusion, followed by surgical bleeding control if necessary, can greatly improve the survival odds of patients that are at risk of entering hemorrhagic shock.

This means that it is essential to activate the procedure as early as possible if a patient's condition requires it. Unfortunately, it is quite hard to evaluate whether a patient is at risk of hemorrhagic shock[41]. While external bleeding (e.g. from a knife wound) is obvious, internal bleeding (usually from blunt trauma) on the other hand is not easily diagnosed visually or using physiological parameters (heart rate, blood pressure, ...) [16]. A full-body scan or at least an ultrasound examination may be needed[40]. This results in a significant delay in the activation of the procedure.

1.1.2 Motivation for an assistance tool

As we mentioned, it is both far from obvious and very important to diagnose a risk of HS early, and doctors often fail to do so without advanced tests: a study [41] showed doctors to have quite limited performance when trying to predict a patient's risk of HS even after 10 minutes in the hospital. This highlights the difficulty of evaluating the need for the MT procedure before arrival, even when a doctor is present in the ambulance.

This combination of factors means that it makes sense to try to provide tools that would assist a doctor in detecting possible HS. To that end, a number of scoring systems have been developed to evaluate the risk of HS for a patient [39] [17] [35]: the idea is to determine a set of conditions on physiological measurements that determine a numeric score for the patient. The idea is that with well-chosen criteria, the score gives an objective assessment of a patient's condition that can supplement a doctor's expertise, or be used directly as a prediction (e.g. if the score is higher than some threshold, then the patient is at risk). However, the same study that evaluated the performance of doctors' prediction [41] showed that numeric criteria perform no better than doctors in predicting HS.

This might mean that it is simply impossible to accurately predict HS before advanced examinations are performed. However, it is also possible, that the relations between HS and physiological measurements are complex enough that a simple hand-made criterion is not enough to capture them. In that case, it would be useful to build a statistical model capable of representing this relationship and of providing hospitals with early estimates of a patient's level of risk.

This is why a team of researchers is trying to develop a tool that would leverage machine learning techniques to predict hemorrhagic shock, using a database of patient records (cf Section 1.2). Our paper is part of that effort, with a specific focus on missing data imputation.

1.2 The Traumabase data

1.2.1 The Traumabase project

1.2.2 Data overview

General information

The Traumabase contained the records of 7477 patients at the time of this work. On recommendation of the doctors we worked with, we removed patients who sustained penetrating injuries such as knife or gunshot wounds — 826 patients — (because the presence of a hemorrhage is obvious to assess in this case) and those who had a cardiac arrest before their arrival in the hospital — 396 patients — because this level of gravity is always enough to justify an emergency procedure. We also excluded patients who were redirected to the trauma centre from another hospital (as opposed to directly by the first responders) — 1102 patients — since this does not correspond to our case of study (prehospital evaluation). This leaves us with a total of 5153 patients.

In this population, 500 patients went through hemorrhagic shock and 4653 did not.

The Traumabase records dozens of variables that trace a patient's history from the moment first responders arrive to the end of the patient's stay in the hospital (i.e. death or recovery). Here we are interested in performing a prehospital evaluation, so when we perform the prediciton we only consider a few measurements that correspond to those performed by the first responders.

Definition of the variables

There are 9 variables in the data that we can use for prediction.

General physical criteria These values are the sex, age and BMI (body-mass index) of the patient. They do not give any direct indication of shock, but they are necessary to control for natural differences between individuals (for instance, males naturally have a higher level of hemoglobin in their blood than females).

Basic physiological measurements These values are measured by the response team as soon as they arrive at the scene. They are:

1. Heart rate: The heart rate of the patient. Intuitively, if the patient has been losing blood, their heart should be beating faster in order to keep supplying the body with oxygen in spite of the blood loss [18]

2. Pulse pressure: The difference between the systolic (maximal) and diastolic (minimal) blood pressure during a heart beat. When the volume of blood is the body is low, this pressure may decrease [18]

3. Hemoglobin level: This is the concentration of hemoglobin (Hb) in the blood. The blood is composed, among other things, of red blood cells which contain hemoglobin that is used to carry oxygen. During blood loss, the liquid part of the blood can be regenerated faster than the red cells [18] which causes a drop in the Hb concentration. It is easily measured on location using measurement kits [32].

4. Peripheral oxygen saturation: This value ranges from 0% to 100% and represents the fraction of Hb molecules in the blood carrying dioxygen. During bleeding, if the oxygen carrying capacity is reduced (lower Hb concentration, lower blood flow due to hypotension, ...) then organs will draw more oxygen relative to the total carrying capacity and the saturation will decrease [12]. Measurement is easy and standard [42].

Glasgow coma scale (GCS) The GCS is a score assessing the conscious state of the patient [26]. It is computed from three criteria (eye movement, verbal response, motor functions) and ranges from 3 (deep coma or death) to 15 (fully awake). It gives a standardized way of reporting a patient's consciousness.

Volume expander injection To stabilise the patient and compensate major fluid loss, the emergency responder may decide to inject the patient with volume expanders [30], that is fluids specifically designed to fill some of the volume of the vascular system in order to rise blood pressure. This is a proxy for the responder's assessment of the patient's gravity, which is useful since many hard-to-quantify factors (paleness, gravity of the incident, general aspect, ...) may impact this assessment and would otherwise be unavailable to us.

This variable gives us the total volume (in mL) of expander that was injected into the patient.

Exploration

Continuous variables: The following table gives a general summary of the continuous covariates:

•	Min	Max	Mean	Median
Age	12	95	37.9	34
BMI	12	100	24.8	24.2
Heart rate	12	222	95.7	93
Pulse pressure	0	169	46.3	45
Hb level	0	19	13.9	14
O2 saturation	0	100	96.5	98
Expander	0	6250	791	500

Their distribution is illustrated in Figure 1.1. We see that all the variables seem to have a unimodal distribution (some variables such as the expander are artificially rounded by the doctors when reporting, which accounts for the apparent drops in density). Additionally, an all variables but the expander dose, the mean and median are very close together.

The age has a rather heavy tail on the right (many old patients). The O2 saturation has a very long lower tail: while almost all patients are above 90% saturation, it is much lower for a few patients.

The differences between the populations of patients with and without shock are in line with our expectations: shocked patients have in general lower Hb levels, a higher heart rate, lower pressure and lower saturation. However, we also see that no single factor gives an easy separation, and that shocked patients can have normal readings for any given measurement.

Categorical variables We have just two variables which can be seen as categorical: the sex and the GCS score. For the GCS, this is a discrete scale from 3 to 15. Its distribution is shown on figure 1.2. As before, shocked patients tend to have a lower score but many of them have a perfect score (15).

As regards the sex, there are 1177 females and 3976 males in the population.

Correlation structure Figure 1.3 shows the correlation between the values of the observations (including the patient outcome). We see that not all variables have obvious correlation, but there are some subgroups of variables that are all correlated (e.g. Glasgow, heart rate, pulse pressure and saturation; or age, BMI and Hb level).

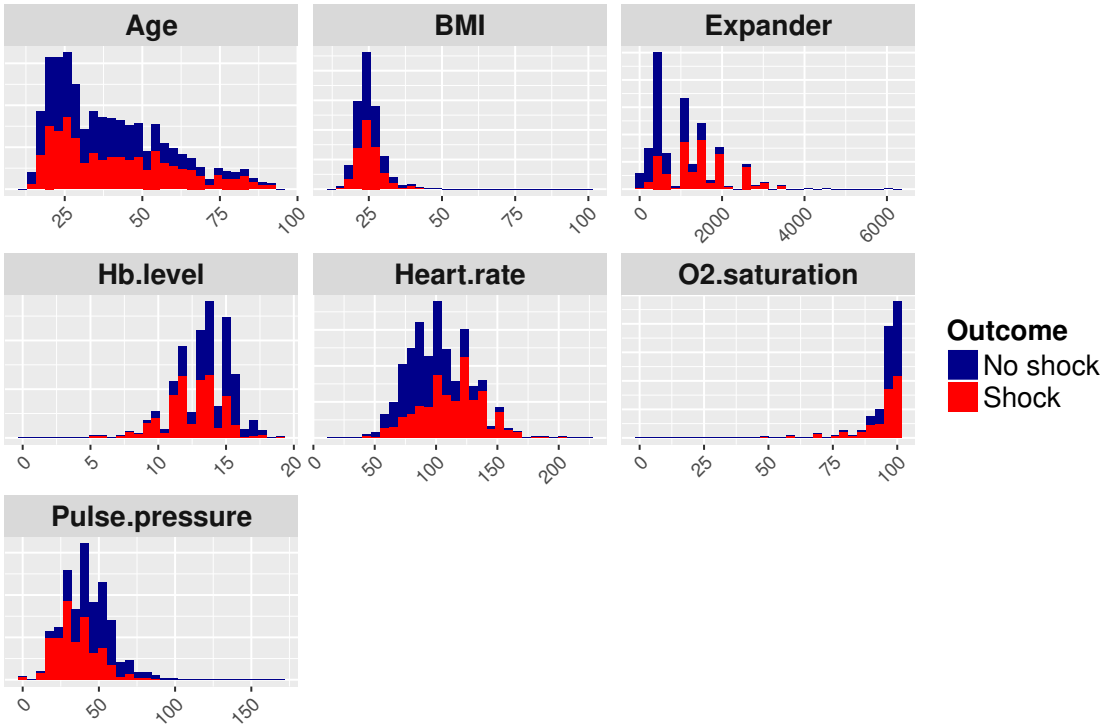


Figure 1.1: Distribution of the continuous variables depending on patient outcome

As expected, the physical measurements (Sex, BMI, age) show no correlation with patient outcome, while all the other variables do.

Correlation with binary variable undefined: use other indocator for association

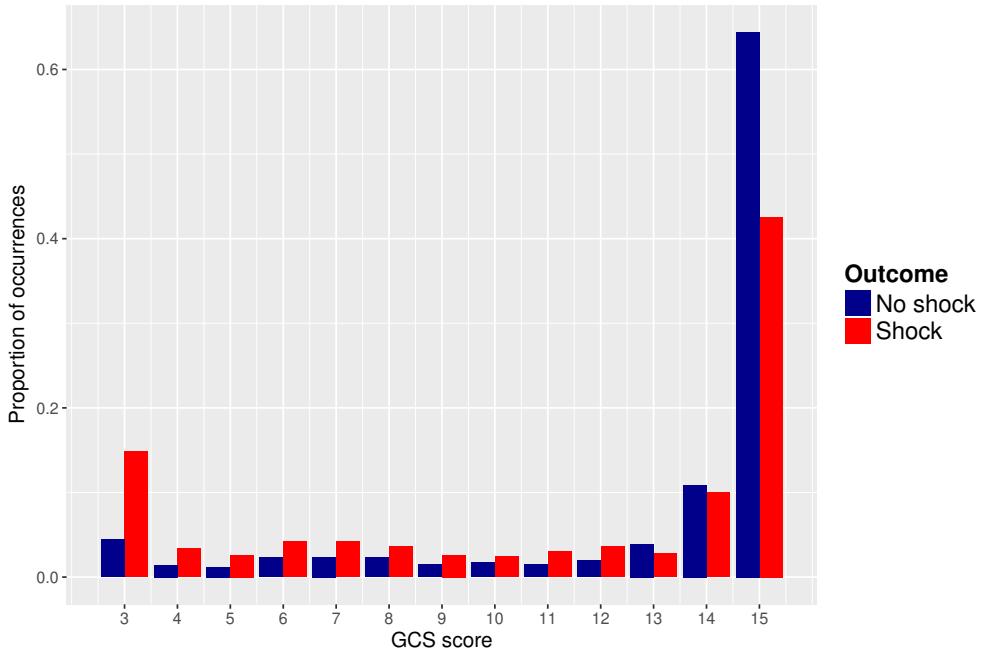


Figure 1.2: Distribution of Glasgow Coma Scale score depending on patient outcome

Sex	1.00	−0.07	0.10	−0.02	−0.04	0.09	0.38	−0.02	−0.03	−0.07
Age	−0.07	1.00	0.23	−0.04	−0.08	0.01	−0.19	−0.10	0.04	0.09
BMI	0.10	0.23	1.00	0.02	0.05	0.02	0.05	−0.05	0.02	0.04
Glasgow	−0.02	−0.04	0.02	1.00	−0.11	0.04	0.10	0.17	−0.22	−0.14
Heart.rate	−0.04	−0.08	0.05	−0.11	1.00	−0.12	−0.03	−0.17	0.22	0.23
Pulse.pressure	0.09	0.01	0.02	0.04	−0.12	1.00	0.11	0.08	−0.20	−0.21
Hb.level	0.38	−0.19	0.05	0.10	−0.03	0.11	1.00	0.04	−0.17	−0.24
O2.saturation	−0.02	−0.10	−0.05	0.17	−0.17	0.08	0.04	1.00	−0.16	−0.12
Expander	−0.03	0.04	0.02	−0.22	0.22	−0.20	−0.17	−0.16	1.00	0.36
Shock	−0.07	0.09	0.04	−0.14	0.23	−0.21	−0.24	−0.12	0.36	1.00

Figure 1.3: Correlation between the measurements (based on complete cases)

PCA

Missing data

An important aspect of the Traumabase is that is contains a significant amount of missing data. That is, some measurements or informations about the patients were not collected or not reported into the database, which makes them unavailable to us. In total, 5% of the observations are missing. The table below gives the amount and proportion of missing data for each variable:

•	Amount	Proportion
Sex	0	0%
Age	7	0.1%
BMI	778	15%
GCS	18	0.4%
Heart rate	110	2.1%
Pulse pressure	126	2.5%
Hb level	301	5.8%
O2 saturation	171	3.3%
Expander	795	15.4%

Figure 1.4 shows the repartition of the number of missing observations in the data. 1686 patients (33 %) have at least one missing observation.

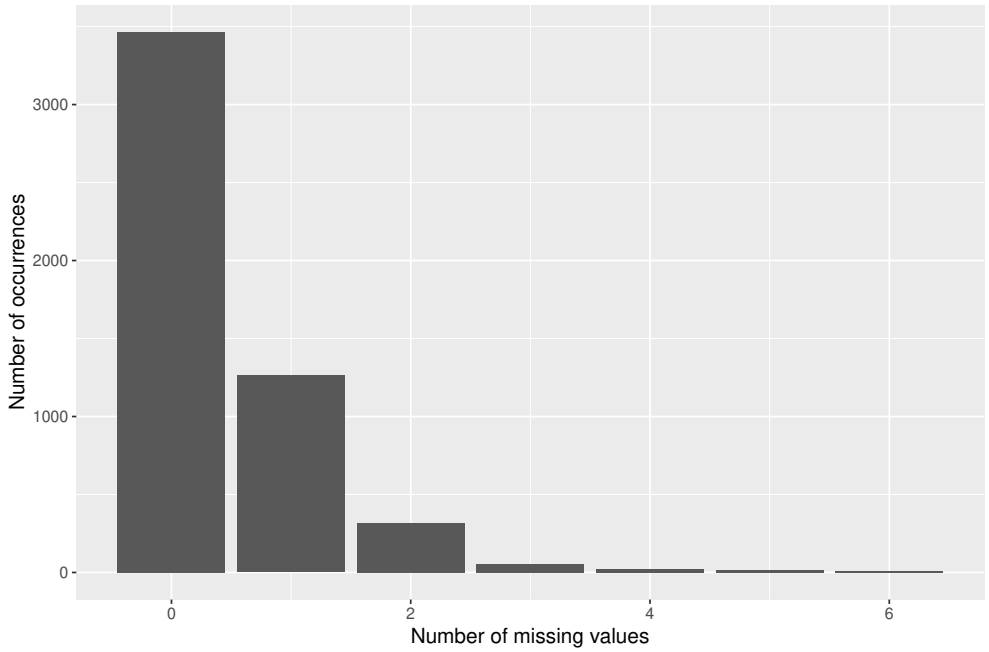


Figure 1.4: Distribution of the number of missing values

Figure 1.5 shows the correlation between the missingness of the variables. They are mostly uncorrelated, but we see that physiological measurements tend to be missing at the same time.

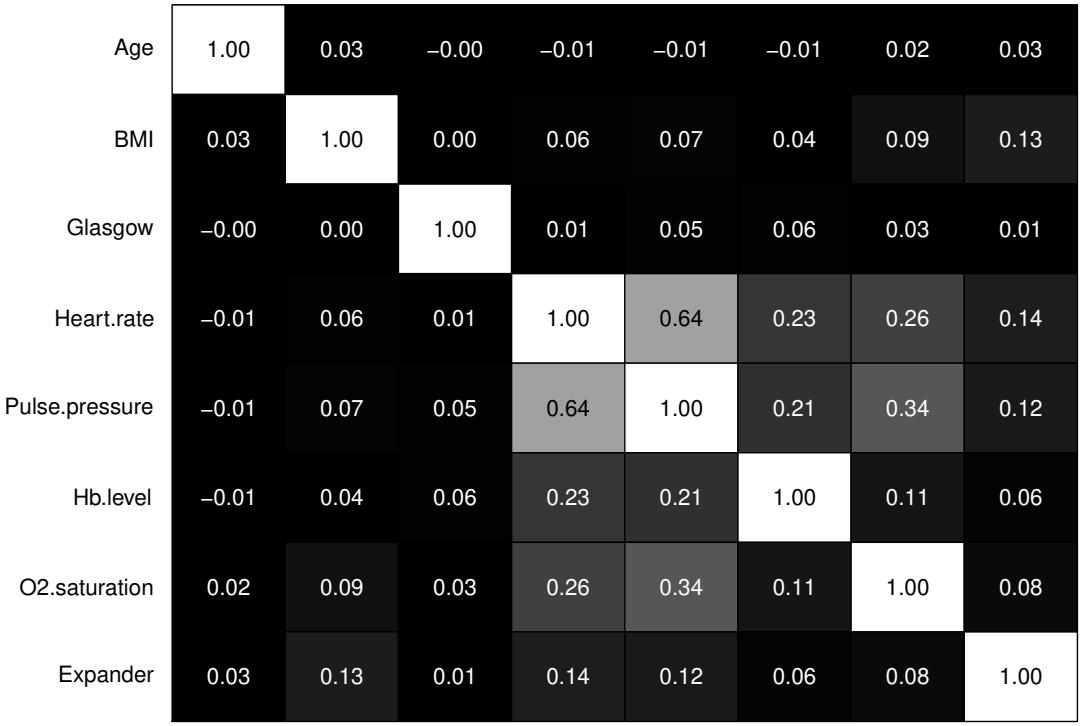


Figure 1.5: Correlation between the missingness of each variable

MCA

Replace correlation with meaningful metric for binary variables

Lastly, the table below gives the correlation between the missingness of each variable with the patient outcome:

Age	$5.7 \cdot 10^{-3}$
BMI	$5.7 \cdot 10^{-3}$
GCS	$2.8 \cdot 10^{-3}$
Heart rate	$-7.6 \cdot 10^{-3}$
Pulse pressure	$5 \cdot 10^{-2}$
Hb level	$-4.5 \cdot 10^{-2}$
O2 saturation	0.11
Expander	$-2.6 \cdot 10^{-2}$

There is some amount of correlation between the missingness of the O2 saturation and the shock, but it is still quite low so it is hard to tell whether this has any significance.

1.3 Objective and formalization

1.3.1 Objective of this work

Given the importance of treating HS quickly, and the difficulty of detecting it — especially for first responders not specialized in major trauma and do not have access to a hospital's equipment —, there is a strong case for trying to predict HS automatically during the prehospital phase. This would enable a hospital to have an assessment of a patient's level of risk as soon as first responders reach them, and make preparations in advance to treat them urgently if necessary.

If one is to develop a tool that would predict hemorrhagic shock, an issue that will need to be addressed is that of missing data. Indeed, there are missing observations in the records of 33% of the patients. Although this leaves us with a rather large number of complete cases, using only those would still be a major loss of information. Even more importantly, some data may also be missing in the real world when a prediction needs to be made. In that case, there no way around handling the missing observations to output a prediction.

In this work, we will address the particular issue of missing data, and more precisely of imputation: replacing the missing values in the dataset by plausible ones. Indeed, it is possible to create a model that takes missing data into account without trying to fill in the missing values and it has been done successfully in the past [47]. However, existing missing-data implementations are fairly rare, so trying out any model means working almost from the ground up. This greatly limits our ability to compare several models.

Comparatively, once the dataset is imputed, it can be used with any existing complete-data prediction method. Additionally, it allows a separation of the tasks: the person or team performing the imputation is not necessarily the same as the one performing the subsequent analysis. Of course, this approach has drawbacks as well. In particular, once the dataset is imputed, any method used to perform a prediction with this dataset will use the observed and imputed values indiscriminately, which may lead to errors if the imputation is not correct.

In the following chapters, we investigate the possibility of performing imputation in a predictive context, and how it should be done. Below we present the formal setting of the problem we investigate.

1.3.2 The God/Imputer/Analyst/Practitioner framework

Let us recall the tasks at hand: imputing the data in the Traumabase, then applying a complete-data procedure to learn a model for the outcome. Finally, for new incoming patients, use the new measurements (possibly with missing data) to evaluate whether they are at risk of HS.

It is clear that form the point of view of the end user (the hospital or medical practitioner), the performance of this procedure should be judged by its predictive performance on new patients. This separation between historical data and new patients is central to our problem, and we investigate it further in chapters 3 and 4.

To formalize this setting, we draw inspiration from the framework proposed by Xie and Meng [55] to explore the issue of imputation. In their work, three actors come into play, God, the Imputer and the Analyst. We adapt this framework by adding a fourth actor, the Practitioner, who represents the end user who is only interested in prediction. Their interaction goes as follows:

- "God" (i.e. nature) generates some data $\tilde{X}$ and outcome y based on a process known only to him.
- A dataset X is generated by adding missing values to $\tilde{X}$. X and y are transferred to an Imputer. The Imputer is tasked with filling in the missing observations. She chooses an imputation model with a parameter α of her choice, and computes an estimate $\hat{\alpha}$ which she uses to generate a completed dataset X_{imp}.
- The Imputer transfers X_{imp} to an Analyst, along with y. The Analyst does not know which observations were initially missing.
- The Analyst uses X_{imp} and y to learn a predictive model with parameter β. Her estimation for this parameter is $\hat{\beta}$
- God generates a new pair of data and outcome $\tilde{X}_{new}, y_{new}$. Some missing data are added to $\tilde{X}_{new}$ to generate X_{new}.
- The Practitioner receives X_{new}. She has no access to the data used for training. The Analyst and Imputer provide the Practitioner with black-box functions that allow her to perform imputation and prediction on the new data. They are derived from their model and parameter estimate; we call them $f(\cdot, \hat{\alpha}), g(\cdot, \hat{\beta})$.

- The Practitioner uses those functions to compute $X_{new}^{imp} = g(X_{new}, \hat{\alpha})$ and $\hat{y}_{new} = f(X_{new}^{imp}, \hat{\beta})$.
- $\hat{y}_{new}$ is finally compared to y_{new} and the loss $L(y_{new}, \hat{y}_{new})$ is computed. This gives the final performance evaluation of the process.

This process is illustrated in Figure 1.6

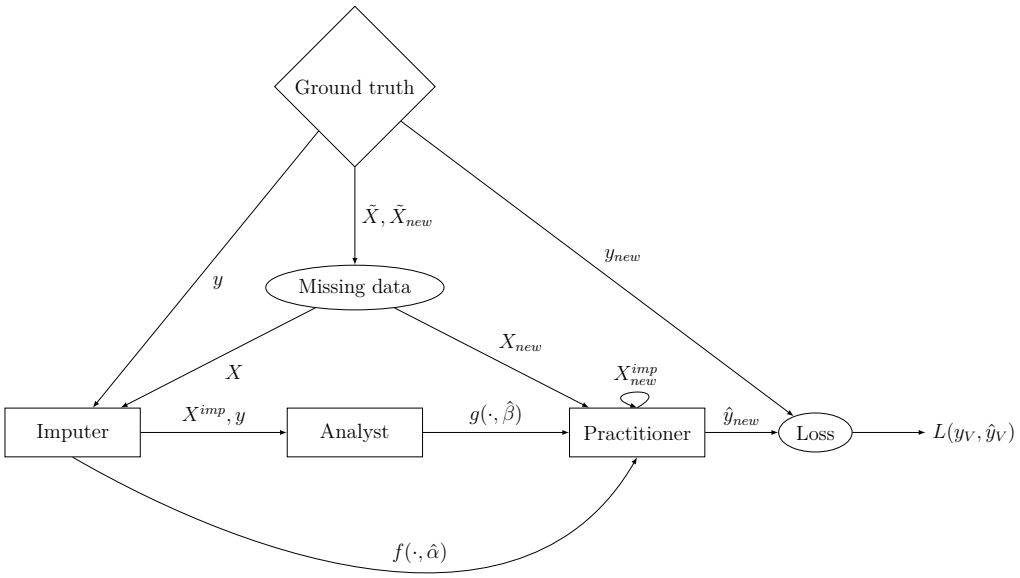


Figure 1.6: Imputation framework

The purpose of this formalization is to clarify the separation between the different phases of the inference, and show how the information is divided. Note that the separation between the Imputer and the Analyst is artificial — we choose it because we are interested in performing imputation separately — while the distinction between the Practitioner and the Analyst is imposed by the problem we are trying to solve: during an intervention, the users need a black-box tool that takes in the available data and outputs a prediction more or less instantly. Even if they had access to the training data, it would be impractical to learn a model all over again every time a new patient comes up. Additionally, the data we are using to learn the model cannot be widely shared since it contains patient records, so the Practitioner will not have access to it.

In a setting where all three roles are regrouped, the sensible way to proceed would be to define a joint model on the full data (including the

response) and use it to find the maximum likelihood estimator for the unknown outcomes. However, the segmentation of the roles makes it necessary for each agent to work with partial information. In the rest of this work, we investigate the implications of this division, both in terms of theory and of practical implementation.

Transition to segue into chapter 2

Chapter 2

Imputation

2.1 Missing data mechanisms

Data may be missing for various reasons, and this leads to various patterns of missing data: in particular, the relationship between the missingness of observations and the true values.

Given some data X, we denote by M the missingness indicator matrix: its coefficients are 1 if the corresponding value is missing in X and 0 otherwise. We call $f(M|X, \phi)$ the distribution of M given the data and some unknown parameters. The missingness patterns can be classified into three categories [34, Ch. 1]:

- *Missing completely at random (MCAR):* The missingness of any given value is independent of the values of the data

$$f(M|X, \phi) = f(M|\phi)$$

- *Missing at random (MAR):* The missingness depends only on the values of X that are observed and not on the missing ones

$$f(M|X, \phi) = f(M|X_{\text{obs}}, \phi)$$

  An example would be a survey on income where we know the age of respondent, and younger people fail to declare their income more often.

- *Missing not at random (MNAR):* The missingness depends on the values that are missing. In the same survey example, this would occur if richer people fail to declare their income more often.

In particular, Rubin showed [43] that if the data is MAR or MCAR, then the likelihood factorises so that maximum-likelihood estimates can be computed by maximizing just the observed likelihood.

2.2 Main types of imputation

2.2.1 Joint maximum likelihood

The most straightforward way to impute data is to assume that the data is distributed according to some parametric joint distribution on all of the variables. In that case, once the distribution has been chosen, one needs to estimate the distribution parameters, and it is then possible to replace missing values by their expected value conditional on the observed data (or draws from this distribution for multiple imputation, see below).[31] [21] [51]

2.2.2 Fully conditional specification (FCS)

In FCS rather than a joint model we define p conditional models $\pi_1, \ldots, \pi_p$ where π_i gives the distribution of variable i conditional on the others. We can then obtain an imputed dataset iteratively using an iterative algorithm [45].

Algorithm 2.1 FCS Algorithm

Input: $X, \pi_1, \ldots, \pi_p$
Output: $\hat{X}$

$\hat{X} \leftarrow$ plausible imputation of the missing data (e.g. mean imputation)
while not converged **do**
  for $i = 1 \ldots p$ **do**
    $X^{(i)} \leftarrow$ the i^{th} column of X
    $\hat{X}^{(-i)} \leftarrow \hat{X}$ without its i^{th} column
    $\hat{X}^{(i)}_{\text{miss}} \sim P(\hat{X}^{(i)}_{\text{miss}} | X^{(i)}_{\text{obs}}, \hat{X}^{(-i)})$
  end for
end while

The interest of this approach is that is can be very flexible when the variables have very different distribution profiles.

2.2.3 Low-rank approaches

An alternative to assuming some distribution for the dataset is to find a low-rank representation of the data and use it to impute unobserved values. Such approaches [27][11][8] are generally based on Principal Component Analysis (PCA)[54], using the iterative PCA algorithm [29]:

Algorithm 2.2 Iterative PCA Algorithm

Input: X, k
Output: $\hat{X}$

1: $\hat{X} \leftarrow$ plausible imputation of the missing data (e.g. mean imputation)
2: **while** not converged **do**
3: $V \leftarrow k$ first principal components of $\hat{X}$ (complete dataset)
4: $\tilde{X} \leftarrow \hat{X}$ projected on the span of V (i.e. PCA fitted values)
5: $\hat{X} \leftarrow \hat{X} * (1 - M) + (\tilde{X} * M$ where M the missingness indicator
6: **end while**

That is, the missing values are repeatedly imputed by projection on the principal components. Many methods using different univariate models exist [45][49][52].

2.2.4 Nearest-neighbors

Other methods exist to impute missing values. *Nearest-neighbor imputation*[10]consists in replacing missing values for one individual by the values observed in similar individuals. Using the observed value, a distance metric is computed with the other observations in the data, and we pick the closest one which has observations in the variable we need to impute. This value is then used for imputation.

The related *hot-deck imputation* [7] use multiple similar individuals to generate imputed values. Rather than pick a single closest value, we can pick a pool of similar individuals and draw at random between their observed values, or impute via a combination of those values.

2.3 Multiple imputation

2.3.1 Principle

One major drawback of imputation is that it hides the difference between values that were really observed and those that were initially missing. If one

uses full-data analysis on an imputed dataset, the results may be overconfident, and in particular underestimate variances because the uncertainty from missing data is not taken into account.

In order to retain the advantages of imputation (using any complete-data method) while compensating for this overconfidence, Rubin [44] introduced *multiple imputation*. The idea is that rather than just one dataset, one should generate m plausible imputed values for each missing observation, in order to account for the uncertainty of the imputation. By performing her inference on each imputed dataset, the Analyst then obtains a set of estimates rather than just one.

When the imputation is based on a distribution, one can use draws from the distribution rather than the expected mean as one would do for single imputation. In other cases, method-specific approaches have to be designed.

Once the datasets are generated, and estimates have been computed, it is possible to combine them to obtain a new estimation for the variances of our estimates[34, Ch. 5].

2.3.2 Rubin's rule for result aggregation

Let us denote $(\hat{\theta}_1, \ldots, \hat{\theta}_m)$ the m estimates for a given parameter θ, and $W_1, \ldots, W_m$ the variance for θ estimated by the complete-data method (within-imputation variance). The aggregated estimate for θ is

$$\hat{\theta} = \frac{1}{m} \sum_{i=1}^{m} \hat{\theta}_i$$

And the aggregated within-imputation variance

$$\hat{W} = \frac{1}{m} \sum_{i=1}^{m} W_i$$

The between-imputation variance can be computed as the sample variance of the estimates:

$$\hat{B} = \frac{1}{m-1} \sum_{i=1}^{m} (\hat{\theta}_i - \hat{\theta})^2$$

Then the total variability associated to θ is then [34, Ch. 5]:

$$\hat{T} = \hat{W} + \frac{m+1}{m} \hat{B}$$

Moreover, if θ is a scalar, asymptotically we can approximate [44]

$$(\theta - \hat{\theta}) \hat{T}^{-\frac{1}{2}} \sim t_\nu$$

where t_ν is a t-distribution with $\nu = (m-1)(1+\frac{1}{m+1}\frac{\hat{W}}{\hat{B}})^2$.

Using this approximation, it is possible to compute confidence intervals for θ that take into account the uncertainty related to missing data.

Chapter 3

Methodology: imputation and the validation split

Let us now go back to the issue of imputing missing data when there is a separation between the Practitioner and the Analyst, that is, when it is necessary to impute new incoming data without having access to the historical data.

Add graphics to explain cv

As described Chapter 2, a number of methods exist to impute missing data. However, they have been designed with parameter estimation in mind: in that case, there is just one dataset that needs to be imputed. The issue, as we show in this chapter, is that current implementations of these methods may not be suitable when we need to use the same model to impute two separate datasets.

This is not only problematic for our particular problem. As we remind in Section 3.1, it is standard practice when working on prediction to resort to CV, that is to hold out part of the data and use it to validate the model. Section 3.2 shows how this clashes with current implementation of imputation methods. In section 3.3, we investigate possible ways to address this issue and compare them empirically.

3.1 Empirical risk minimization and cross-validation

Let us start by ignoring the issue of missing data and assume that the data is complete. That is, we place ourselves in the same situation as described in 1.3.2 but there is no need for an Imputer, and the Analyst directly receives the data X.

As stated in Chapter 1, our end goal is to make good predictions in the real world by learning on historical data. Of course, by definition we do not have access to any future data right now, but we still need to choose a prediction and imputation model, and estimate how it will perform when we use it on the field.

To learn and evaluate the model, we go through two steps: Empirical risk minimization (ERM) and cross-validation (CV). We describe them below

3.1.1 Empirical risk minimization

Let us denote the X a $n \times p$ matrix of covariates and y a response vector of size n, where our goal is to predict the response y_{new} from some future data X_{new}, assuming that (X_{new}, y_{new}) follow the same distribution as X, y.

We choose a class of functions $f(\cdot, \beta), \beta \in B$. We want to choose a parameter $\hat{\beta}$ which we can use to predict $\hat{y}_{new} = f(X_{new}, \hat{\beta})$. The quality of a prediction is evaluated by some loss $L(y_{new}, \hat{y}_{new})$. Since we do not have access to X_{new}, our goal is to minimize the risk: $R(\beta) = \mathbb{E}_{X_{new}, y_{new}}(L(y_{new}, f(X_{new}, \beta)))$.

However, we do not know the true distribution of those values. This is why we resort to ERM [53]: we define the empirical risk

$$R_{\text{emp}}(\beta) = \frac{1}{n} \sum_{i=1}^{n} L(y_i, f(X_i, \beta))$$

that is, the average value of the loss when predicting the known y from X with β.

We then select $\hat{\beta} = \arg\min_\beta R_{\text{emp}}(\beta)$ as our ERM estimator for β. However, this is not enough. Once we have chosen $\hat{\beta}$, we need to have an estimate of how well this estimate will perform on new data.

This is important because this is what we will take into account if we need to compare multiple choices for f. But the empirical risk gives us no measure of how well our model generalizes, only of how closely it can fit known data. In particular, if the class f is very broad, one may find a β that exactly interpolates the values of y but does not generalize at all (an issue known as overfitting [19]

To address the issue of model selection, we need to resort to CV as descried below.

3.1.2 Cross-validation

CV, consists in dividing the available data in two datasets: first we choose $n_A < n$ entries in the dataset that will be used in ERM to learn $\hat{\beta}$: this is

the training dataset X_A and response y_A. We denote $I_A = (i_1, \ldots, i_{n_A})$ the set of indices chosen for the training data.

The rest of the observations are noted X_V and y_V and called the validation dataset. They are used as a substitute for X_{new}, y_{new}

Once this is done, the Analyst performs ERM as before, using only the training data. The obtained parameter $\hat{\beta}$ can then be evaluated with the validation error:

$$R_V(\hat{\beta}) = \frac{1}{n_V} \sum_{i=1, i \notin I_A}^{n} L(y_i, f(X_i, \hat{\beta}))$$

It is this value that we can compare to choose the model class f. Once the Analyst has decided on a choice of f and $\hat{\beta}$ using ERM and CV, she can send $f(\cdot, \hat{\beta})$ over to the Practitioner so that she can proceed to prediction on new data using $\hat{y_{new}} = f(X_{new}, \hat{\beta})$.

3.2 ERM with missing data: the problem of current methodologies

We now place ourselves in the same context as before, except some values are missing from X, both in the training and the validation data. This means that we are back to a case where there is an Imputer in addition to the Analyst.

3.2.1 Imputation seen as an ERM

Remember that the purpose of this work is to impute the data independently of the model used afterwards for prediction. This means that we cannot perform ERM exactly as before and use any function we like to go from X (which has missing data) to $\hat{y}$. The prediction is the composition of two steps.

Imputation step First we choose an imputation model $X^{\text{complete}} = g(X, \alpha)$ where X^{complete} is the completed dataset and α some parameter. This is similar to the previously described ERM, except we do not know the true data (while we had y to compare to $\hat{y}$, we do not know the true full dataset $\tilde{X}$). Thus, we choose $\hat{\alpha}$ to minimize some unsupervised empirical risk

$$R'_{emp}(\alpha) = L'(g(X, \alpha), \alpha)$$

Sometimes the loss L' is related to the likelihood of the completed data according to some distribution[21] (though it is not always the case [49]). Once this is done, we obtain a completed dataset $\hat{X}$.

Prediction step With imputation done, we can proceed as before to choose a parameter $\hat{\beta}$ that minimizes the empirical risk when using the completed data:

$$R_{\text{emp}}(\beta, \hat{X}) = \frac{1}{n} \sum_{i=1}^{n} L(y_i, f(\hat{X}_i, \beta))$$

Putting it all together, we can define

$$h(X, (\alpha, \beta)) = f(X^{imp}, \beta) = f(g(X, \alpha), \beta)$$

the combined model that takes the observed data as input and outputs a predicted y. Formally, the two successive steps yield:

$$\begin{aligned} \hat{\alpha} &= \arg\min_{\alpha} L'(g(X, \alpha) \\ \hat{\beta} &= \arg\min_{\beta} R_{\text{emp}}(\beta, f(X, \hat{\alpha})) \\ \hat{y} &= h(X, (\hat{\alpha}, \hat{\beta})) \end{aligned}$$

We choose to use this notation to illustrate our point that imputation is an integral part of the ERM, not a separate, preliminary process. In particular, it means that its parameters must be subjected to CV just like those of the prediction. That is, only X_A and y_A are used to estimate $(\hat{\alpha}, \hat{\beta})$ as shown above, while X_V and y_V are held out. Then, we can compute a prediction $\hat{y}_V = h(X_V, (\hat{\beta}, \hat{\alpha}))$ and we compute $L(y_V, \hat{y}_V)$ to evaluate the choice of model.

This implies that just like β, the imputation parameter α should be estimated only on the training data and then used on the validation data. As we will see, this raises an issue with the way current imputation methods are implemented.

3.2.2 Unsuitability of current methods

Implementations of imputation methods have one thing in common: they are used through a single function which takes a dataset with missing values as an input and returns the dataset completed by the method of choice, *without giving the user any access to the imputation model itself.* [49] [28][45][21]

This is a problem because of how CV is supposed to be performed. Supposedly, one would estimate $(\hat{\alpha}_A, \hat{\beta}_A)$ through ERM, and then make a prediction on the validation set as $h(X_V, (\hat{\beta}_{X_A}, \hat{\alpha}_{X_A}))$. But here, all we have access to is a black-box function $g' : X \mapsto g(X, \hat{\alpha}_X)$ where $\hat{\alpha}_X$ is the optimised parameter for the argument X. This means that one cannot choose what parameters are used to impute the input to function g': a new parameter will be estimated at every call of the function. But in that case, it is impossible to use the same α for the training dataset and for the validation data — or for new data.

This issue has started to arise in the machine learning community in the past few years [2][4][3], but for now no implementation exists that separates the parameter estimation and the imputation itself (except for the very basic imputation by the mean [5]). Below, we investigate the alternatives available to perform imputation with held out data.

3.3 Possible solutions

If we were to follow exactly the principles of CV, we would proceed as follows:

Algorithm 3.1 Identical imputation

Input: X, y, $I_A = \{i, X_i \in X_A\}$
Output: $\hat{y}_V$

1: **Parameter estimation:**
2: $\hat{\alpha}_A \leftarrow \arg\min_\alpha L'(g(X_A, \alpha)$
3: $\hat{X}_A \leftarrow g(X_A, \hat{\alpha}_A)$
4: $\hat{\beta}_A \leftarrow \arg\min_\beta R_{\text{emp}}(\beta, \hat{X}_A)$
5: **Prediction:**
6: $\hat{X}_V \leftarrow g(X_V, \hat{\alpha}_A)$ ▷ *Uses the same α as for the training set*
7: $\hat{y}_V \leftarrow f(\hat{X}_V, \hat{\beta}_A)$

But this is not possible using a black-box function because we need to recover $\hat{\alpha}_A$ and reuse it with X_V.

3.3.1 Alternatives using current implementations

Motivation

As mentioned above, many implementations of imputation methods only provide a black-box function. If we are to use one of these methods to

build a tool for hemorrhagic shock, we need to be able to impute on a new incoming patient, without any access to the training data or to other new patients' data, so we have no choice but to use past estimated parameters as in identical imputation. This necessitates a new implementation of the same method.

However, before building a final model for real-world use, there is a phase of model selection that is done using CV, where we actually have access to all of the data. Reimplementing a method is a time-consuming process, so we would prefer to be able to choose the best imputation method and reimplement just this one, rather than having to remake every method we are interested in just to compare them.

This is why we explore options that can be used to impute the training and validation CV datasets, using only a black-box function. In particular, we try to see if those methods give a good idea of a method's true performance, and so would allow us to perform model selection with existing implementations.

Methods

Grouped imputation Impute all the data at once before performing the CV split:

Algorithm 3.2 Grouped imputation

Input: X, y, $I_A = \{i, X_i \in X_A\}$
Output: $\hat{y}_V$

Imputation:
  $\hat{X} \leftarrow g'(X) = g(X, \hat{\alpha}_X)$
  $(\hat{X}_A, \hat{X}_V) \leftarrow \hat{X}$ ▷ *Split the data after imputation*
Estimation of the prediction parameter
  $\hat{\beta}_A \leftarrow \arg\min_\beta R_{\text{emp}}(\beta, \hat{X}_A)$ ▷ *Note that* $\hat{X}_A = g(X_A, \hat{\alpha}_X)$
Prediction:
  $\hat{y}_V \leftarrow f(\hat{X}_V, \hat{\beta}_A) = f(g(X_V, \hat{\alpha}_X)$

Here, both datasets are indeed imputed with the same parameter α_X but this means that the validation data is used to choose that parameter which then serves to impute the training data. This is contrary to the principles of CV, we are 'cheating' in some way.

Separate imputation Divide the data first, then impute each dataset separately:

Algorithm 3.3 Separate imputation

Input: X, y, $I_A = \{i, X_i \in X_A\}$
Output: $\hat{y}_V$

Training parameter estimation:
  $\hat{X}_A \leftarrow g'(X_A) = g(X_A, \hat{\alpha}_A)$
  $\hat{\beta}_A \leftarrow \arg\min_\beta R_{\text{emp}}(\beta, \hat{X}_A)$
Imputation of X_V:
  $\hat{X}_V \leftarrow g'(X_V) = g(X_V, \hat{\alpha}_V)$ ▷ *Imputation made independently of that of X_A*
Prediction:
  $\hat{y}_V \leftarrow f(\hat{X}_V, \hat{\beta}_A)$

Contrarily to grouped imputation, we are not cheating. However, we are using parameter $\hat{\alpha}_V$ to impute X_V, while we learned $\hat{\beta}_A$ on $\hat{X}_A$ which was imputed with $\hat{\alpha}_A$. That is, we are optimising $h(\cdot, (\hat{\alpha}_A, \hat{\beta}_A))$ and predicting with $h(\cdot, (\hat{\alpha}_V, \hat{\beta}_A))$.

$\hat{\alpha}_V$, $\hat{\alpha}_A$ and $\hat{\alpha}_X$ are asymptotically the same for large n — since X_A, X_V and X have the same distribution —, so all three methods should be identical for large n. But for smaller n, harmful effects may be present — overoptimistic validation due to 'cheating' for the grouped imputation, high error due to the difference in parameters for separate imputation.

Need for a new implementation

We want to understand if the alternatives proposed here are good enough to be used if Identical imputation in infeasible. To do that, we need to be able to compare these with the correct method. That means that for at least one imputation method we need to build an implementation that allows us to separate the estimation and the imputation. That way we will be able to compare its performance with the other alternatives we propose.

Moreover, in addition to this theoretical pursuit, we need this because of what we are trying to achieve with Traumabase: the end goal is to make a recommendation system that can produce a prediction for *a single new patient* arriving to the hospital, without needing to have access to the whole Traumabase data. Without access to the initial training data, this means that only a fully parametric approach can be taken in this particular case (separate imputation is impossible on just one line of data, and grouped imputation requires access to the full data).

Below, we design a very simple imputation method for those purposes.

3.3.2 Multivariate normal conditional expectation

The principle of this imputation is inspired from R package *Amelia* [21], and a large part of the code is from the *norm* package [51]. The idea is to model both X_A and X_V as normally distributed $\mathcal{N}(\mu, \Sigma)$ with unknown parameters. This will allow us to impute the missing data conditionally on the observed data.

Parameter estimation It is possible to approximate maximum-likelihood estimators for μ and σ with an iterative procedure, using the EM (expectation-maximisation) algorithm [14] [46]. The algorithm is as follows:

Algorithm 3.4 Normal parameter estimation with EM

Input: X
Output: $\hat{\mu}, \hat{\Sigma}$

$X^{(0)} \leftarrow X$ where missing values are replaced using the observed mean of X (mean imputation)
$t \leftarrow 0$
while not converged **do**
    $(\mu^{(t)}, \Sigma^{(t)}) \leftarrow$ sample mean and covariance matrix of $X^{(t)}$ ▷ *i.e. maximum likelihood estimates on the completed data*
    $X^{(t+1)} \leftarrow \mathbb{E}_{X^{\text{miss}}}(X|X^{\text{obs}}; \mu^{(t)}, \Sigma^{(t)})$ ▷ *Missing values are replaced by their expected value under the new parameters*
    $t \leftarrow t + 1$
end while
$\hat{\mu}, \hat{\Sigma} = \mu^{(t)}, \Sigma^{(t)}$

The conditional expectations are easily derived using the Schur complement [57]. For this step, we use a slightly modified version of the code from the *norm* package. In all that comes next, unless specified otherwise this imputation method is the one we use, with identical imputation.

Imputation Once we have the parameters, it is very straightforward to get an imputation of the missing data. Just as during the EM procedure, we impute using the conditional expectation of the dataset conditioned on the observed values:

$$\hat{X} = \mathbb{E}_{X^{\text{miss}}}(X|X^{\text{obs}}; \hat{\mu}, \hat{\Sigma})$$

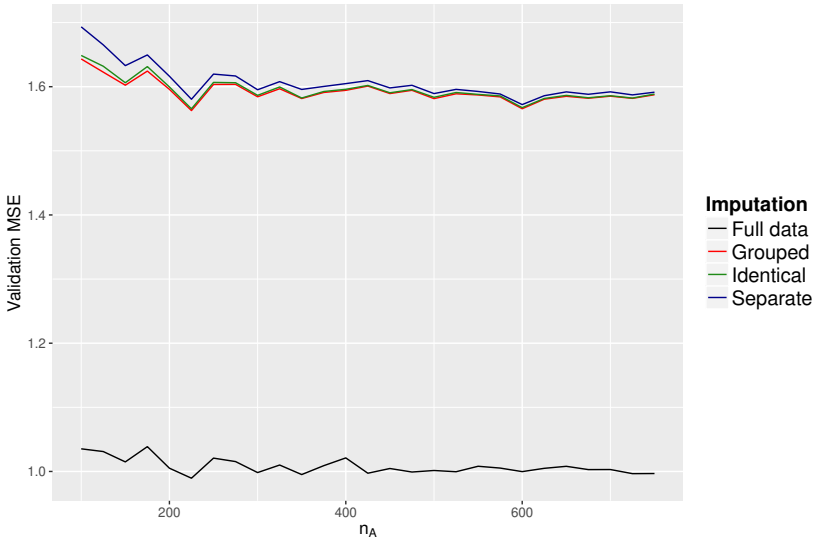
We implemented this step as well to get a complete imputation procedure divided in two functions that implement the estimation and imputation steps separately. The code is available in package

make package and insert reference

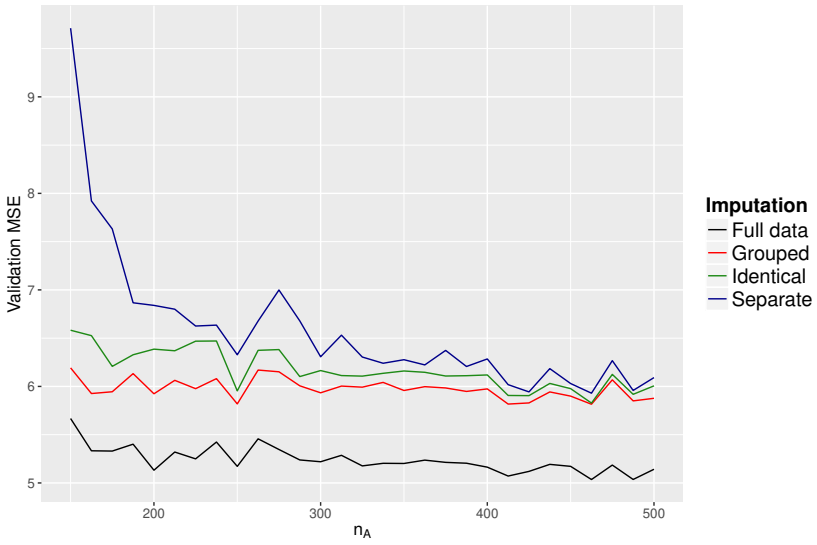
3.3.3 Comparison on simulated data

Now that we have an implementation that separates estimation and imputation, we use it to compare the three imputation procedures on simulated data (cf Appendix A, with $\rho = 0.5, p = 4, \sigma = 1$) and the abalone data (cf Appendix B) with various sample sizes and adding 30% missing data MCAR. The results are in Figure 3.1.

We see that indeed grouped imputation seems to always have lower error than identical imputation, while separate imputation has higher error. However, when n is not too small, all three are quite close together. Separate imputation seems to be further off than the other two, and sometimes has very high error. As a result, if no implementation exists for a given method to perform identical imputation, trying it out with grouped imputation would be a good first step to get an idea of how it can perform.



(a) Results for simulated data



(b) Results for abalone data

Figure 3.1: Comparison of imputation methodologies (averaged over 100 runs)

Chapter 4

Impact of missing data: the case of linear regression

In order to make good decisions for imputation, it is important to understand how it impacts prediction. To gain a better understanding of this issue, we solve a very simple case of cross-validated regression with missing data. Although quite restrictive, this situation provides some insights into the way that missing data impacts prediction performance. In particular, we want to understand what makes this situation different from one of pure parameter estimation (i.e., without a Practitioner), and the implications of missing data in a prediction context.

We show that these objectives are compatible, in the sense that if the imputer performs a good imputation while the analyst uses a good full-data procedure in some sense, then the predictive loss is asymptotically good. In addition, this example makes it clear that the presence of missing values in the validation dataset means that the situation does not boil down to parameter estimation.

4.1 Problem set-up

We place ourselves in a linear regression setup with cross-validation (cf Chapter 3). The data is split between a training dataset X_A, y_A and validation dataset X_V, y_V. We use the multi-agent framework described in 1.3.2.

4.1.1 Notations

God's data

The response variable is a noisy linear combination of the covariates in X:

$$\tilde{X}_A = \begin{pmatrix} x_{11} & x_{12} \\ \vdots & \vdots \\ x_{n1} & x_{n2} \end{pmatrix} \quad \text{and} \quad y_A = X_A\beta + \epsilon_A \quad \text{with} \quad \epsilon_A \sim \mathcal{N}(0, \sigma^2)$$

$$\tilde{X_V} = \begin{pmatrix} x^V_{11} & x^V_{12} \\ \vdots & \vdots \\ x^V_{n_V 1} & x^V_{n_V 2} \end{pmatrix} \quad \text{and} \quad y_V = X_V\beta + \epsilon_V \quad \text{with} \quad \epsilon_V \sim \mathcal{N}(0, \sigma^2)$$

Observed data

The observed data is God's data with some missing values. Specifically, some observations are missing from the first column of each dataset. We observe the full y^A, but the covariate matrices we actually have access to are:

$$X_A = \begin{pmatrix} ? & x_{12} \\ \vdots & \vdots \\ ? & x_{k_A 2} \\ x_{(k_A+1)1} & x_{(k_A+1)2} \\ \vdots & \vdots \\ x_{n1} & x_{n2} \end{pmatrix}$$

which is sent to the Imputer, and

$$X_V = \begin{pmatrix} ? & x^V_{12} \\ \vdots & \vdots \\ ? & x^V_{k_V 2} \\ x^V_{(k_V+1)1} & x^V_{(k_V+1)2} \\ \vdots & \vdots \\ x^V_{n_V 1} & x^V_{n_V 2} \end{pmatrix}$$

which is sent to the Practitioner. That is, there are k_A and k_v missing values in the datasets (the mechanism is MCAR).

Note that the datasets have a different status. The training dataset is available to the Imputer then the Analyst at the time of analysis, it is some given historical data. The validation dataset is some future data available only to the Practitioner who will perform a black-box prediction based on

the Analyst's and the Imputer's indications. That is why when we take expectations in this chapter, we will condition only on the observed data X_A while we integrate on $X_V, \epsilon_A, \epsilon_V$ and the missing data $X_A^{\text{miss}}, X_V^{\text{miss}}$ which are all unknowns at the time of analysis.

4.1.2 Imputed data and regression

Principle

The Imputer fits an imputation model $g(\cdot, \alpha)$ and fills in X_A and instructs the Practitioner on how to impute X_V. The resulting filled-in datasets are:

$$\hat{X}_A = \begin{pmatrix} g(x_{12}, \hat{\alpha}) & x_{12} \\ \vdots & \vdots \\ g(x_{k_A 2}, \hat{\alpha}) & x_{k_A 2} \\ x_{(k_A+1)1} & x_{(k_A+1)2} \\ \vdots & \vdots \\ x_{n1} & x_{n2} \end{pmatrix} \quad \text{and} \quad \hat{X}_V = \begin{pmatrix} g(x_{12}^V, \hat{\alpha}) & x_{12}^V \\ \vdots & \vdots \\ g(x_{k_V 2}^V, \hat{\alpha}) & x_{k_V 2}^V \\ x_{(k_V+1)1}^V & x_{(k_V+1)2}^V \\ \vdots & \vdots \\ x_{n_V 1}^V & x_{n_V 2}^V \end{pmatrix}$$

Then, $\hat{X}_A$ is sent by the Imputer to the Analyst. The Analyst only has access to $\hat{X}_A$ and y_A. The end goal is to learn an estimator on the training set that minimizes the expected loss on the validation set:

$$L(y_V, \hat{y}_V) = (y_V - \hat{y_V})^2$$

In line with the principles of ERM and CV (cf Chapter 3), the Analyst minimizes the equivalent quantity in the training set. Assuming a linear relationship between the covariates and response, the least-squares estimate for β is standard [48]

$$\hat{\beta}_n = (\hat{X}_A^T \hat{X}_A)^{-1} \hat{X}_A^T y_A$$

$\hat{\beta}$ is then transferred to the Practitioner who can use it to compute a prediction

$$\hat{y}_V = \hat{X}_V \hat{\beta}_n$$

which will be compared to y_V:

$$L(\hat{y}_V, y_V) = \sum_{i=1}^{n_V} (y_V^{(i)} - \hat{y}_V^{(i)})^2$$

Our end goal is to minimise this metric.

In what we described above, the actions of the Analyst and the Practitioner are completely determined. On the other hand, we have not specified

how the Imputer proceeds to the imputation. We want to investigate the effect of the choice of imputation on the expected loss:

$$R = \mathbb{E}_{\tilde{X}_V, X_A^{miss}, \epsilon_A, \epsilon_V}[(y_V^{(i)} - \hat{y}_V^{(i)})^2 | X_A]$$

Distribution hypotheses

Lastly, for this last expression to have any meaning, fix the distribution of $\tilde{X}$ the true data.

We assume that $X \sim \pi$ where the lines of X are independent and identically distributed (i.i.d), and that π is known to the Imputer. This is unlikely in a real setting, but we explore the best-case scenario where we have all the necessary information to perform the imputation in order to isolate the error terms that are specific to the presence of missing data — as opposed to bad imputation.

4.2 Analysis

4.2.1 Expected loss

To be able to estimate the expected loss, we break it up into several components. We first denote

$$\tilde{\beta}_n = (\tilde{X}_A^T \tilde{X}_A)^{-1} \tilde{X}_A^T y_A$$

the estimated parameter we would obtain if the training data were completely observed. We consider the loss for the i^{th} line of validation data x_i^V:

$$
\begin{aligned}
L_i(y_V, \hat{y}_V) &= (y_V - \hat{y_V})^2 \\
&= (\tilde{x}_i^V \beta + \epsilon_V - \hat{x}_i^V \hat{\beta}_n)^2 \\
&= (\tilde{x}_i^V(\beta - \tilde{\beta}_n) + \tilde{x}_i^V(\tilde{\beta}_n - \hat{\beta}_n) + (\tilde{x}_i^V - \hat{x}_i^V)\hat{\beta}_n + \epsilon_V)^2 \\
&= (\tilde{x}_i^V(\beta - \tilde{\beta}_n))^2 && (1) \\
&\quad + (\tilde{x}_i^V(\tilde{\beta}_n - \hat{\beta}_n))^2 && (2) \\
&\quad + ((\tilde{x}_i^V - \hat{x}_i^V)\hat{\beta}_n)^2 && (3) \\
&\quad + 2\tilde{x}_i^V(\beta - \tilde{\beta}_n)\tilde{x}_i^V(\tilde{\beta}_n - \hat{\beta}_n) && (4) \\
&\quad + 2\tilde{x}_i^V(\beta - \tilde{\beta}_n)(\tilde{x}_i^V - \hat{x}_i^V)\hat{\beta}_n && (5) \\
&\quad + 2\tilde{x}_i^V(\tilde{\beta}_n - \hat{\beta}_n)(\tilde{x}_i^V - \hat{x}_i^V)\hat{\beta}_n && (6) \\
&+ \epsilon_V^2 \\
&+ \epsilon_V C
\end{aligned}
$$

Where C is some term that will not matter (since it will not count in the expectation — ϵ_V has zero expectation and is independent of the other terms). We can see that terms $(1), (2)$ and (4) depend only on the imputation of the training values ($\hat{x}^V$ is absent), while term (3) depends only on the imputation of the validation data ($\hat{\beta}_n$ is absent). Terms (5) and (6) show the interaction between both imputations.

Influence of missing validation values In particular,

Proposition 4.1. *The expected validation error depends linearly on the proportion of missing data in the validation set (with a fixed amount of missing data in the training set).*

Proof. The risk we want to minimize is R_i the expectation of this loss, summed over all rows. Since the rows are i.i.d, (so the order does not count), the expected value of terms $(1), (2), (4)$ is the same for all i. For terms $(3), (5), (6)$ there are two possibilities: if there is no missing data in the row, these terms are zero. If a value is missing, they are nonzero but their expectation is the same for all lines with missing data.

Consequently, denoting $r_V = \frac{k_V}{n_V}$ the proportion of missing validation data, we can express the expected loss as:

$$\sum_{i=1}^{n_V} L_i = \underbrace{\mathbb{E}_{\tilde{X_V}, X_A^{miss}, \epsilon_A, \epsilon_V}[(1)+(2)+(4)|X_A]}_{A} + \tag{4.1}$$

$$r_V \underbrace{\mathbb{E}_{\tilde{X_V}, X_A^{miss}, \epsilon_A, \epsilon_V}[(3)+(5)+(6)|X_A]}_{B} + \sigma^2 \tag{4.2}$$

Thus, for X_A fixed and for a given imputation rule, the expected loss is $A + \sigma^2 + Br_V$ with A and B fixed, and the expected loss depends linearly on the proportion of missing values. □

Consistency In linear regression without missing data, the estimation of the parameter is consistent[6], that is in our case $\tilde{\beta}_n$ is a consistent estimate of β. Moreover, Little [33] studied parameter estimation with missing values and showed:

Proposition 4.2. *If the missing data is MCAR and the imputed values are the expected values of the unobserved data conditioned on the observed data ($\hat{x} = \mathbb{E}[x|x_{obs}$), then the least-square estimator $\hat{\beta}_n$ is consistent for β.*

Note that to do this means the Imputer must know the distribution of X, in order to find the expected values.

If $\tilde{beta}, \hat{\beta}$ are consistent, then all terms but $(3) + \epsilon_V^2$ in the loss tend to 0 when $n \to \infty$. Additionally, term (3) is zero for lines without missing data, and for lines with missing data its expectation is:

$$\mathbb{E}[((\tilde{x}_i^V - \hat{x}_i^V)\hat{\beta}_n)^2] = \mathbb{E}[(\hat{\beta}_n^{(2)})^2]\mathbb{E}[(x_{i2}^V - \hat{x}_{i2}^V)^2]$$

Which is minimized by the choice of imputation $\hat{x}_{i2}^V = \mathbb{E}[x_{i2}^V]$, i.e. the same that ensures a consistent estimator for β.

4.2.2 Consequences

These results highlight the asymmetry between training and validation data. In a very simple case, it gives us an intuition of the behaviour of the prediction error with missing data.

We see that when the imputation is done well, consistency results ensure that in the training data, it is ultimately possible to make up for the unobserved values and have an arbitrarily good parameter estimate, if we have enough observations.

On the other hand, missingness in the validation set adds some inevitable terms to the error. This is important because although training and validation data have the same distribution by hypothesis, in some cases the missingness could be somehow different between those datasets.

For instance, in the Traumabase it is possible that missing data comes from doctors who fail to record some values in the database, although the data was available when treating the patient. In that case when real-world prediction are made, there may be less missing values than there were when we performed CV and model selection. On the contrary, maybe some values were collected late in the process and would have been unavailable for the prehospital diagnostic. Proposition 4.1 shows that this matters a lot, and that efforts to reduce the amount of missing data at the time of real-world predictions may have a more direct effect on performance than efforts to reduce missing data in the database (e.g. by improving recording).

Lastly, presence of term (3) even for large n and consistent estimation also shows that the intrisic variability of the covariates imposes a limit on the performance of the prediction with missing data (while otherwise the only limit would be linked to the noise ϵ in the response).

Still, we see that in this case at least, what could seem like independent goals (imputation and full-data parameter estimation) are actually compatible: if we try to minimize error term that depends only on imputation (using imputation by the conditional mean), then we are also enabling consistent parameter estimation which makes the other error terms tend to zero.

In the next chapter, we take a more empirical approach to investigate similar issues.

Chapter 5

Imputation and prediction: Empirical findings

5.1 Impact of missing data

5.1.1 Is less missing data always better?

When performing an analysis, it is intuitive that we should limit the amount of missing data as much as possible, since missing data pollutes our estimates.

Ajouter une erreur de référence: au meilleur de la prédiction, on est bons?

In particular, if the missing data is MCAR — and so the complete cases have exactly the same distribution as those with missing data —, and we have a large enough dataset with many complete cases (as in the Traumabase), it is tempting to use only those complete cases to learn our model. Even in a context where we are training for prediction, and the real-world data will have some missing values we need to handle, it seems that we can use complete cases in the training data to learn both our prediction and imputation parameters as accurately as possible and then use those to predict the new data at best.

However, it may not be so: when imputing missing data, we do not recover the exact initial data. What if these errors change the structure of the dataset enough that a different parameter (possibly different from the one that generated the data) can yield better predictions? In that case, learning our model without any missing data may yield the true parameter but still not be optimal for prediction.

We investigated this on some simulated (cf Appendix A with $n = 4000, p = 5, rho = 0.9, \sigma = 10$) and real-world data (abalone, cf Appendix B) by adding a fixed proportion of missing values to the validation data and varying the amount of missing data in the training set:

Algorithm 5.1 Impact of missing data

Input: $\pi_V, m, X_A, X_V, y_A, y_V$
Output: $L_1, \dots L_m$

1: **for** $\pi_A \in [0, \frac{1}{m}, \dots \frac{m-1}{m}]$ **do**
2: Add proportion π_A of MCAR missing data to X_A
3: Add proportion $\pi_V A$ of MCAR missing data to X_V
4: Impute $\hat{X}_A$ and $\hat{X}_V$ using μ_A the observed mean of X_A
5: Compute $\hat{\beta}_A$ by linear regression on $\hat{X}_A, y_A$
6: Predict $\hat{y}_V = \hat{X}_V \hat{\beta}_A$
7: $L_i \leftarrow L(\hat{y}_V, y_V)$
8: **end for**

The results are shown in Fig.5.1 (where π_{min} indicates the point where the lowest error is achieved).

What we can see here is that when the training dataset is fully observed while the validation has missing data, the prediction error is *higher* than in the same situation with missing data in the training set as well. More precisely, the best prediction is achieved when both datasets have roughly the same amount of missing data.

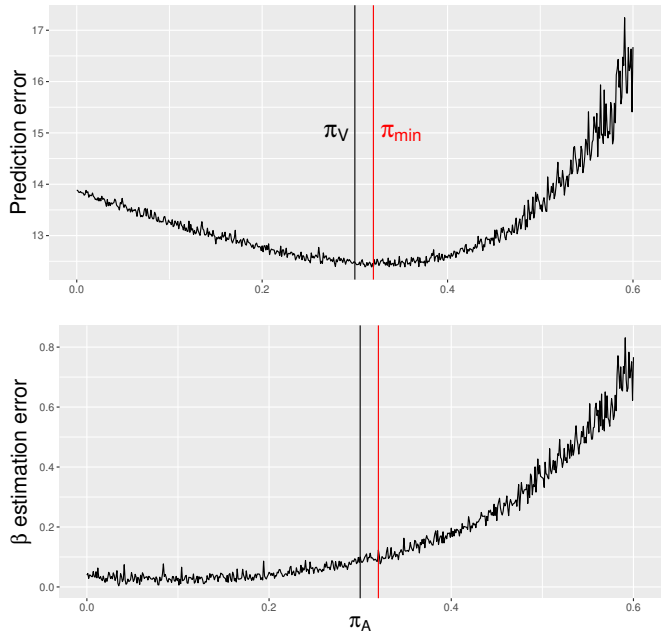
It is not clear how general this result is. In particular, we could obtain it *only when using mean imputation*: with more elaborate imputation methods it did not show.

In any case, this warrants caution when doing cross-validation with missing data: while reducing the amount of missing data in our records is a worthy endeavour — e.g. by deleting incomplete cases —, it is possible that it will only be useful if the real-world (or validation) data also has less missing data as a result — e.g., improving the data-collection process. Additionally, just as it is important to ensure that the distribution of the data is stable between training and application — no temporal trend in the data —, the same should be done about the missing-data pattern.

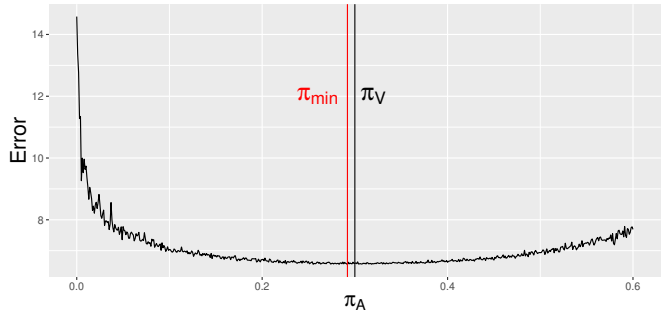
5.1.2 Asymmetry between the two datasets

We want to keep investigating an observation from the previous chapter: missing data does not have the same impact on performance when it is in the training set as when it is in the validation set, and depending on the situation one or the other may be determinant of the value of the error.

To do this we simulate data in the same fashion as above, that is with a normal X and a linearly derived response y. For various proportions π, we



(a) Prediction and parameter estimation errors for simulated data



(b) Prediction error for abalone data

Figure 5.1: Impact of missing data in the training set

then perform imputation and prediction in 4 different cases:

- Proportion π of MCAR missing values in both datasets: we note the loss L_B
- Proportion π of MCAR missing values just in the validation set: we note the loss L_V
- Proportion π of MCAR missing values just in the training set: we

note the loss L_A

- Fully observed data: we note the loss L_F.

Figure 5.2 shows the results of this process, for simulated (cf Appendix A with $p = 45, rho = 0.5, \sigma = 1$) and real-world data (abalone, cf Appendix B), adding 30% MCAR data . We observed that the variable that caused the most change in the relationship between each type of error was p, the number of covariates, which is why we choose to present the results for different values of p. We can see that L_B tends to follow the trend imposed by either L_A or L_V, whichever is worse: when p is small, L_B is almost equal to L_V (i.e., with few parameters to estimate the estimation of β is good so the biggest impact is from the imputation error in X_V). When p is larger, L_B starts following the trend of L_A (with many parameters to estimate, the error on β causes very large errors to appear), although L_B stays smaller than L_A, which is the same effect that we showed in 5.1.1.

5.2 Multiple imputation

Actually implementation is wrong, results to be fixed

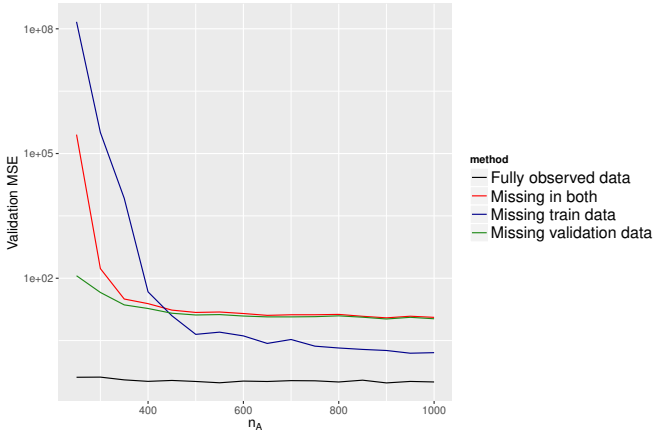
Instead of making only one imputation, it is possible (see Chapter 2) to instead impute multiple datasets and perform the analysis on each dataset. Just like for parameter estimates, we can generate multiple predictions and use these to estimate prediction uncertainty from missing data.

5.2.1 Partial multiple imputation

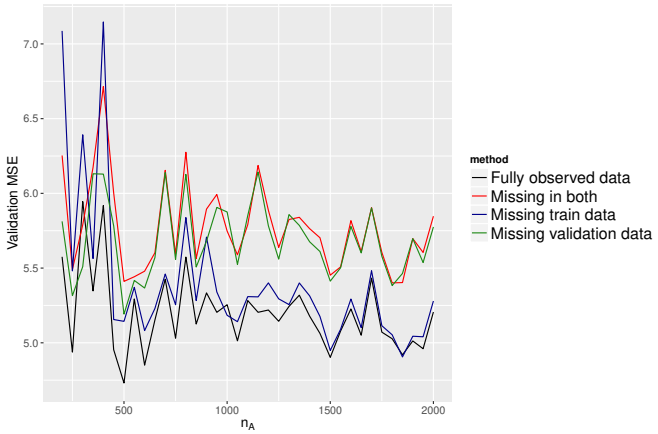
In Chapter 4, we mention the fact that if the estimation is consistent, then most of the prediction uncertainty is linked to missing data in the validation dataset, and not to that in the training one. This suggests an idea that would allow predictive multiple imputation while keeping the computational cost rather low.

Given some training data X_A and validation X_V, we can impute m datasets $X_A^{(1)}, \dots X_A^{(m)}, X_V^{(1)}, \dots X_V^{(m)}$ which would yield estimates $\hat{\beta}_n^{(1)}, \dots, \hat{\beta}_n^{(m)}$ and then $\hat{y}_V^{(1)}, \dots, \hat{y}_V^{(1)}$. However, this implies making m separate parameter estimations. If n is large, the variance of $\hat{\beta}$ may be insignificant in the variance of $\hat{y}_V$ compared to the direct impact of missing validation data.

This is why we can try instead to impute the training dataset only once, while imputing the validation data m times as usual, and use the same $\hat{\beta}$ to predict on each imputed validation set. This would be a major computational gain, since the most computing-intensive part of the analysis is usually parameter estimation, not prediction.



(a) Results for simulated data (log scale)



(b) Results for abalone data

Figure 5.2: Impact of missing data in the train and validation set

To see is this is worth considering, we perform an analysis on some very simple simulated data.

5.2.2 Prediction intervals

As in 5.1.1 we generate X normal covariates and y a noised linear combination of these, then split them for CV. We add the same amount of MCAR missing data in both datasets, then impute them multiply (both as described just above and by only imputing the training data once). We train a linear regression on X_A, y_A and predict on X_V.

This allows us to compute the between-imputation variance for each

entry in $\hat{y}_V$, and the linear regression model gives us an estimation for the within-imputation variance of the predictions. We can then use Rubin's rule (cf Chapter 2) to compute a total variance for each prediction, and the t-distribution approximation to obtain an interval. Figure 5.3 shows average coverage rates for $p = 20, \rho = 0.5, m = 30$, 30% missing data and multiple values of n (averaging over several runs). We see that when n is very small, partial MI tends to severely underestimate variance while full MI overestimates it. When n is larger, both perform similarly and close to nominal coverage.

explain within-imputation variance calc

Note, though, that we could not reproduce those results even on simple real-world data: coverage was vastly inferior to nominal values. It is plausible that unless the model fits the data perfectly, the estimated intervals can be very misleading.

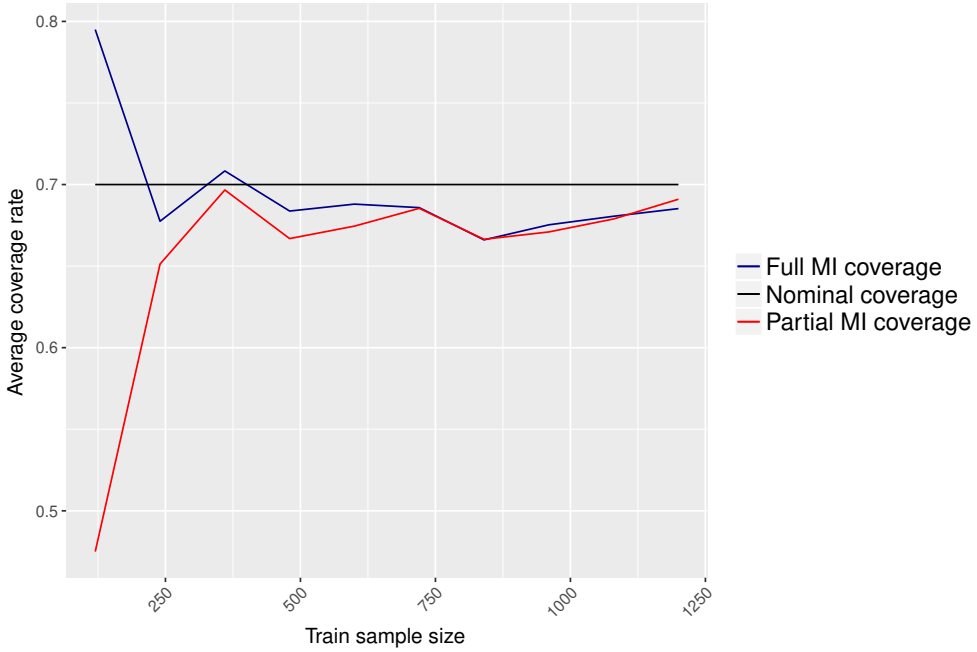


Figure 5.3: Mean coverage rate of prediction intervals

Chapter 6

Imputing the Traumabase data for prediction

Our final step is to see how well prediction with imputation works to predict hemorrhagic shock. For that, we proceed to cross-validation (cf Chapter 3) to get an idea of the performance we can achieve this way. We compare it to several references to see whether this is an improvement over other methods of prediction.

6.1 Methodology

6.1.1 Prediction pipeline

To evaluate our method we perform imputation and prediction on the data, by training on a training set X_A, y_A and validating on X_V, y_V. In order to get an idea of the average performance, we repeat this for multiple X_A, X_V splits.

An interest of imputation is that once missing data is imputed, any full-data method can be used for prediction. To illustrate this, after imputation we perform prediction using three different methods: logistic regression [22], Support Vector Machine (SVM) [20] and random forest (RF) [50].

6.1.2 Evaluating the prediction

Metric The response variable in our data is binary, and we predict a probability. This, and the unbalance in the response (only 10% of positive cases) means that the choice of metric is not straightforward. A first choice we have to make is whether we choose a threshold for the prediction (predict

a positive when the predicted probability is above some value), or evaluate the predicted probability as-is.

Some metrics allow us to evaluate the predicted probability directly, such as the AUC [24] or log-loss (minimized by the logistic regression). However, we want to be able to compare our results with those given by scores or the historical decisions of doctors, which are binary. We want to see if our predictions are able to separate patients with and without shocks at least as well as those references, so we need a metric that puts our predictions and those scores on an equal footing.

To that end, we choose a simple cost function that, given a binary prediction, assigns some user-defined cost to false negatives and false positives. That is:

$$L(\hat{y}, y) = \frac{1}{n} \sum_{i=1}^{n} c_1 \mathbb{1}_{y_i=1,\hat{y}_i=0} + c_2 \mathbb{1}_{y_i=0,\hat{y}_i=1}$$

with $c_1 + c_2 = 1$.

To evaluate our predicted probability, we take the best value of this loss for any choice of threshold: this gives us a measure of the separation power of those predictions. The choice of costs is not obvious, which is why in this chapter we show the results for multiple possible values.

Comparison In order to have a reference performance for HS prediction, we compare the value of the loss with the loss obtained from several other predictions:

- *Doctor's prediction:* The decision to initiate a MT procedure is recorded in the Traumabase. It determines whether the doctor considered the patient to be at risk of HS.
- *ABC (Assessment of Blood Consumption)[39] score:* this gravity score is the only one that was designed with prehospital prediction in mind. It is a very simple score that only uses a few measurements.
- *TASH (Trauma Associated Severe Hemorrhage)[56]) score:* this score was also designed for hemorrhage detection, but at a later stage: it uses some values that are only available after laboratory tests (e.g. base excess) or radiography (presence of a fracture).
- *SAEM Logistic regression:* this a method for logistic regression without imputation of the training dataset, developed by Jiang [25] to address the specific issue of HS prediction on the Traumabase. If we take the notations from Chapter 3, the idea is that α the imputation parameter

and β the regression parameter are learned jointly rather than one after the other.

This gives us some points of comparison for predictive performance.

6.1.3 Choice of imputation method

There are many methods of imputation we can choose from (cf Chapter 2) to impute missing values in. In order to compare them, we proceed to grouped imputation (as described in Chapter 3) with each of them, and then perform a prediction on each imputed dataset. The resulting validation errors are presented in figure 6.1.

We used the following imputation methods (c.f. Chapter 2):

- *Mean imputation:* replace missing values by the observed mean of the corresponding column
- *Normal expectation:* impute by approximating the data as multivariate normal and taking conditional expectations (c.f. Chapter 3)
- *PCA imputation:* impute through a low-rank approximation of the data
  FCS with adapted methods: FCS with a normal regression model for all variables except the Sex (logistic regression) and GCS (proportional odds model for ordered variables).
- *MissForest:* a FCS method that uses random forests as the univariate predictor

Note that the Sex is binary and the GCS is ordered discrete, so we cannot in theory impute them with the PCA or normal imputation. However, de Sex has no missing data and the GCS has only a handful (0.4%) so we proceed with the imputation by treating them as numerical variables.

It is striking that the difference between imputation methods is very small: even though there are differences in the mean performance, these differences are manor compared to the variation of the performance for different CV splits. Even imputation by the mean, which is supposedly very inaccurate, is on par with other methods in terms of performance. We were not able to determine the reason for this: the relatively low proportion of missing data does not seem to be in cause as the same trend shows if we artificially add missing values.

In what follows, the results are presented for normal imputation.

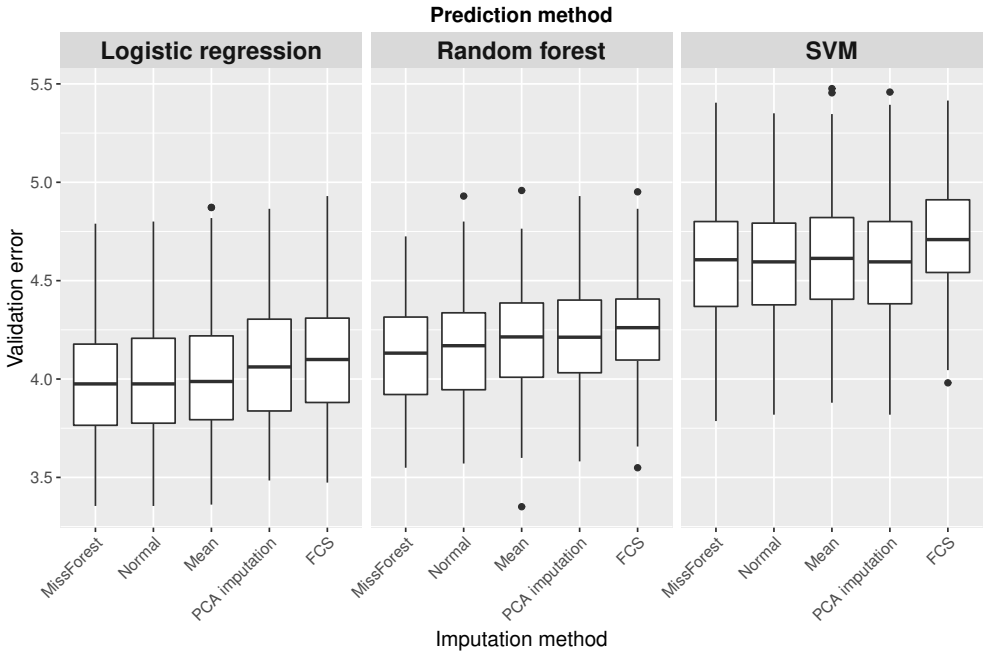


Figure 6.1: Prediction performance for multiple prediction and imputation methods

6.2 Results

We performed predictions on the Traumabase data as described above, for 16 different CV splits. Figure 6.2 shows the average loss of each prediction (ours and the reference values) for different values of $\frac{c_1}{c_2}$.

First note that there are two possible main trends for the loss depending on the method:

- The loss increases with c_1: this means that the predictions are more conservative, and tend to have fewer false positives but more false negatives. This is the case of the doctors' prediction and the ABC score.

- The loss decreases when c_1 increases: this means that the prediction tends to favor overpredicting HS, so it has fewer false negatives but more false positives. This is the case of all our predictions, as well as the TASH score and SAEM prediction.

When two predictions follow the same trend, it is easy to compare them as one is usually above the other regardless of the choice of weights. In

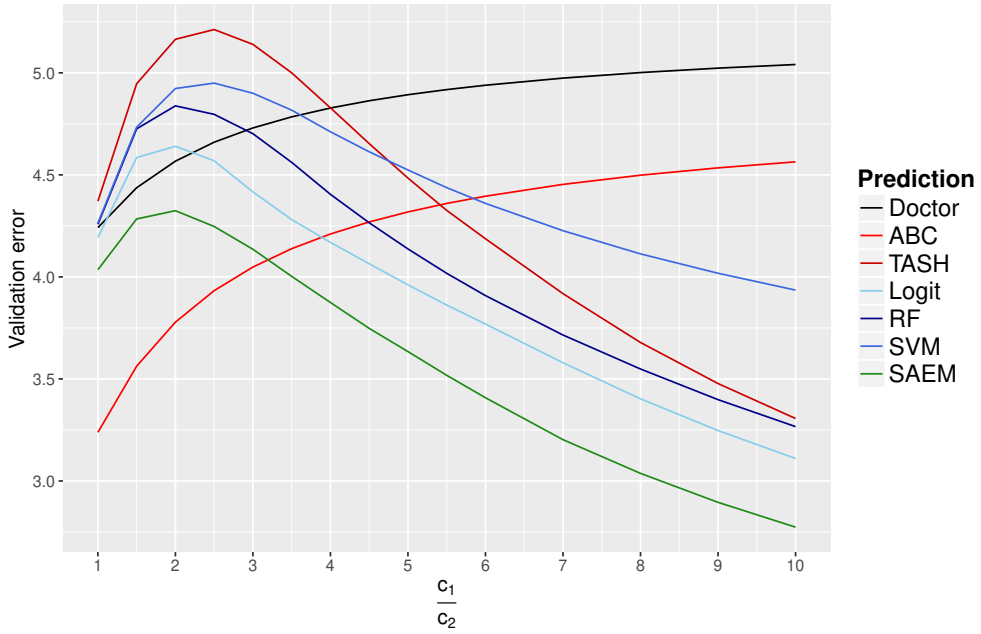


Figure 6.2: Error for each prediction depending on c_1, c_2

particular, the ABC score seems to be strictly better at predicing hemorrhage than doctors, even though it is extremely simple in principle.

Likewise, we see that for the descending trend, the best predictor is without a doubt the SAEM regression, followed by the logistic regression with imputation, the random forest with imputation, the TASH score and the SVM with imputation in that order.

Whether the ABC score should be preferred to other predictors depends on the choice of weights: for the SAEM prediction, the loss is lower than for ABC if $\frac{c_1}{c_2}$ is greater than approximately 3, while for the regression with imputation this happens for a ratio above 4.

All in all, what this shows is that although imputation gives promising results, for a given model there is a tangible advantage to performing a joint prediction rather than separating imputation and prediction.

Still, imputing is a good first step: thanks to grouped imputation and subsequent prediction, we can evaluate a method's promise using only out-of-the box methods, without having to create a dedicated model. For instance, we can see that it is visibly not worth trying to implement a model for SVM with missing data. If for instance the performance on the imputed data had been better for RF than for logistic regression, it would have given us a strong case for building a missing-data random forest as a next step.

Model selection using imputed datasets allows us to single out one method that we can then choose to improve by implementing it to work without imputation.

Conclusion

In this work, we investigated the possibility to predict hemorrhagic shock in trauma patients when imputing the missing values, both in the database and at the time of prediction. We tried to understand how this taks should be performed, and found that to use existing imputation methods in the real world, they must be modified so that data for a single new patient can be imputed (as opposed to a block of many lines). We implemented such a method and compared it to other alternatives that can be used for model selection and to not require a new implementation, and found that grouped imputation seemed to be a good indicator of a methods true performance.

We also studied a simplified case to understant the specificities of our task. It is clear that the missing values in the validation data/new patients means that estimating a good regression parameter on the training data is not our only task (we also need to choose how to impute the new data). In favorable cases (with MCAR data), we show that these two objectives go hand in hand: when the imputation is done using the true conditional expectation and the estimation uses a consistent full-data method on the imputed data, then the imputation error is minimal and the parameter estimation remains consistent on the imputed data.

Additionally, if we have some control over the amount of missing data on the training data or the real world data, but not both (which can happen, as the causes of missingness can be separate: a measurement can be made by the doctor and be available at the time of diagnostic, then not be recorded into the base. On the contrary, a measurement can be delayed because of lack of time or instruments and be completed once the patient reaches the hospital), then one should focus on limiting misingness in the real world data because missing information at the time of prediction reflects much more directly in the validation error. On the contrary, decreasing only the amount of missing training data may actually have a negative effect when it comes to predicting new data with missing values.

The final evaluation of imputation on the Traumabase data gave us mixed results. Indeed, it appears that the joint SAEM regression gives significantly

better results than prediction after imputation. Still, performing imputation allowed us to compare many prediction methods with minimal efforts, which had the potential to guide us in our future choices of imputation.

In the future, it would be worthwile to quantify formally the gap in predictive performance between joint optimization, and imputation followed by prediction with the same hypotheses (for instance, SAEM has the same distribution hypotheses as normal imputation followed by logistic regression: it is only the choice of parameters that differs). In the investigation of HS prediction, an important next step would be to understand why all imputation methods seem to perform similarly.

Appendix A

Simulated normal data

To simulate a regression dataset with normally distributed covariates, we proceed as follows.

Algorithm A.1 Data simulation

Input: $n_A, n_V, p, \rho, \sigma$
Output: X_A, X_V, y_A, y_V

1: $\Sigma \leftarrow (1-\rho)I_p + \rho \mathbb{1}_{p\times p}$ ▷ Same correlation ρ between all pairs of variables
2: $X_A \sim \mathcal{N}(0, \Sigma)$ ▷ Of size $n_A \times p$
3: $X_V \sim \mathcal{N}(0, \Sigma)$ ▷ Of size $n_V \times p$
4: $\beta \leftarrow (1, \ldots, 1)$
5: $y_A \leftarrow X_A\beta + \epsilon_A$ ▷ Where $\epsilon_A \sim \mathcal{N}(0, \sigma^2)$
6: $y_V \leftarrow X_V\beta + \epsilon_V$ ▷ Where $\epsilon_V \sim \mathcal{N}(0, \sigma^2)$

That is, for both the training and validation data, we generate a normally distributed X and a response y that is a linear combination of X with normal noise.

Appendix B

Abalone data

In addition to simulated data, we want to make some tests on real-world data. The Traumabase data is not adapted for this use, because it has missing data (while we want a dataset where we know the full data, in order to add missing values ourselves and see the effects). In addition, it has a binary response which makes it harder to evaluate the prediction.

Instead, we use the Abalone [38] dataset. It consists in measurements on abalone (sea snails) shells where the goal is to predict their age. It has 4177 observations, 7 numerical covariates and one categorical covariate. We keep only the numerical ones. The dataset is fully observed.

In addition, there is a strong correlation between all the pairs of variables (> 0.8).

Bibliography

[1] Cause-specific mortality and morbidity - World Health Organization. http://www.who.int/whosis/whostat/EN_WHS09_Table2.pdf.

[2] Get at the final model used in the MICE iterations? https://github.com/stefvanbuuren/mice/issues/32. Github MICE repository.

[3] Imputation using MICE: Use the train data to impute the missing test data. https://stats.stackexchange.com/questions/332342/imputation-using-mice-use-the-train-data-to-impute-the-missing-test-data. Stackexchange discussion.

[4] R MICE impute new observations. https://stackoverflow.com/questions/40115226/r-mice-impute-new-observations. Stackoverflow discussion.

[5] Scikit-learn imputer. http://scikit-learn.org/stable/modules/generated/sklearn.preprocessing.Imputer.html.

[6] TW Anderson, John B Taylor, et al. Strong consistency of least squares estimates in normal linear regression. *The Annals of Statistics*, 4(4):788–790, 1976.

[7] Rebecca R Andridge and Roderick JA Little. A review of hot deck imputation for survey non-response. *International statistical review*, 78(1):40–64, 2010.

[8] Matthew Brand. Incremental singular value decomposition of uncertain data with missing values. In *European Conference on Computer Vision*, pages 707–720. Springer, 2002.

[9] Jeremy W Cannon. Hemorrhagic shock. *New England Journal of Medicine*, 378(4):370–379, 2018.

[10] Jiahua Chen and Jun Shao. Nearest neighbor imputation for survey data. *Journal of Official statistics*, 16(2):113, 2000.

[11] Pei Chen and David Suter. Recovering the missing components in a large noisy low-rank matrix: Application to sfm. *IEEE transactions on pattern analysis and machine intelligence*, 26(8):1051–1063, 2004.

[12] Stephen M Cohn, Avery B Nathens, Frederick A Moore, Peter Rhee, Juan Carlos Puyana, Ernest E Moore, and Gregory J Beilman. Tissue oxygen saturation predicts the development of organ dysfunction during traumatic shock resuscitation. *Journal of Trauma and Acute Care Surgery*, 62(1):44–55, 2007.

[13] C Clay Cothren, Ernest E Moore, Holly B Hedegaard, and Katy Meng. Epidemiology of urban trauma deaths: a comprehensive reassessment 10 years later. *World journal of surgery*, 31(7):1507–1511, 2007.

[14] Arthur P Dempster, Nan M Laird, and Donald B Rubin. Maximum likelihood from incomplete data via the em algorithm. *Journal of the royal statistical society. Series B (methodological)*, pages 1–38, 1977.

[15] Richard P Dutton. Current concepts in hemorrhagic shock. *Anesthesiology clinics*, 25(1):23–34, 2007.

[16] Matthias Frank, Uli Schmucker, Dirk Stengel, Lutz Fischer, Joern Lange, Rico Grossjohann, Axel Ekkernkamp, and Gerrit Matthes. Proper estimation of blood loss on scene of trauma: tool or tale? *Journal of Trauma and Acute Care Surgery*, 69(5):1191–1195, 2010.

[17] Eduardo Gonzalez, Ernest E Moore, Hunter B Moore, Michael P Chapman, Theresa L Chin, Arsen Ghasabyan, Max V Wohlauer, Carlton C Barnett, Denis D Bensard, Walter L Biffl, et al. Goal-directed hemostatic resuscitation of trauma-induced coagulopathy: a pragmatic randomized clinical trial comparing a viscoelastic assay to conventional coagulation assays. *Annals of surgery*, 263(6):1051, 2016.

[18] Guillermo Gutierrez, HDavid Reines, and Marian E Wulf-Gutierrez. Clinical review: hemorrhagic shock. *Critical care*, 8(5):373, 2004.

[19] Douglas M Hawkins. The problem of overfitting. *Journal of chemical information and computer sciences*, 44(1):1–12, 2004.

[20] Marti A. Hearst, Susan T Dumais, Edgar Osuna, John Platt, and Bernhard Scholkopf. Support vector machines. *IEEE Intelligent Systems and their applications*, 13(4):18–28, 1998.

[21] James Honaker, Gary King, Matthew Blackwell, et al. Amelia ii: A program for missing data. *Journal of statistical software*, 45(7):1–47, 2011.

[22] David W Hosmer Jr, Stanley Lemeshow, and Rodney X Sturdivant. *Applied logistic regression*, volume 398. John Wiley & Sons, 2013.

[23] David B Hoyt, Eileen M Bulger, M Margaret Knudson, John Morris, Ralph Ierardi, Harvey J Sugerman, Steven R Shackford, Jeffery Landercasper, Robert J Winchell, and Gregory Jurkovich. Death in the operating room: an analysis of a multi-center experience. *The Journal of trauma*, 37(3):426–432, 1994.

[24] Jin Huang and Charles X Ling. Using auc and accuracy in evaluating learning algorithms. *IEEE Transactions on knowledge and Data Engineering*, 17(3):299–310, 2005.

[25] Wei Jiang, Julie Josse, and Marc Lavielle. Saem for logistic regression with missing data.

[26] Cathy Jones. Glasgow coma scale, 1979.

[27] Julie Josse and François Husson. Handling missing values in exploratory multivariate data analysis methods. *Journal de la Société Française de Statistique*, 153(2):79–99, 2012.

[28] Julie Josse, François Husson, et al. missmda: a package for handling missing values in multivariate data analysis. *Journal of Statistical Software*, 70(1):1–31, 2016.

[29] Henk AL Kiers. Weighted least squares fitting using ordinary least squares algorithms. *Psychometrika*, 62(2):251–266, 1997.

[30] George C Kramer. Hypertonic resuscitation: physiologic mechanisms and recommendations for trauma care. *Journal of Trauma and Acute Care Surgery*, 54(5):S89–S99, 2003.

[31] Jonathan Kropko, Ben Goodrich, Andrew Gelman, and Jennifer Hill. Multiple imputation for continuous and categorical data: Comparing joint multivariate normal and conditional approaches. *Political Analysis*, 22(4):497–519, 2014.

[32] Lionel Lamhaut, Roxana Apriotesei, Xavier Combes, Marc Lejay, Pierre Carli, and Benoît Vivien. Comparison of the accuracy of noninvasive hemoglobin monitoring by spectrophotometry (sphb) and hemocue®

with automated laboratory hemoglobin measurement. *Anesthesiology: The Journal of the American Society of Anesthesiologists*, 115(3):548–554, 2011.

[33] Roderick JA Little. Regression with missing x's: a review. *Journal of the American Statistical Association*, 87(420):1227–1237, 1992.

[34] Roderick JA Little and Donald B Rubin. *Statistical analysis with missing data*, volume 333. John Wiley & Sons, 2014.

[35] M1 Maegele, R Lefering, A Wafaisade, P Theodorou, S Wutzler, P Fischer, B Bouillon, T Paffrath, and Trauma Registry of the Deutsche Gesellschaft für Unfallchirurgie (TR-DGU). Revalidation and update of the tash-score: a scoring system to predict the probability for massive transfusion as a surrogate for life-threatening haemorrhage after severe injury. *Vox sanguinis*, 100(2):231–238, 2011.

[36] Debra L Malone, John R Hess, and Abe Fingerhut. Massive transfusion practices around the globe and a suggestion for a common massive transfusion protocol. *Journal of Trauma and Acute Care Surgery*, 60(6):S91–S96, 2006.

[37] Matthew Martin, John Oh, Heather Currier, Nigel Tai, Alec Beekley, Matthew Eckert, and John Holcomb. An analysis of in-hospital deaths at a modern combat support hospital. *Journal of Trauma and Acute Care Surgery*, 66(4):S51–S61, 2009.

[38] Warwick J Nash, Tracy L Sellers, Simon R Talbot, Andrew J Cawthorn, and Wes B Ford. The population biology of abalone (haliotis species) in tasmania. i. blacklip abalone (h. rubra) from the north coast and islands of bass strait. *Sea Fisheries Division, Technical Report*, (48), 1994.

[39] Timothy C Nunez, Igor V Voskresensky, Lesly A Dossett, Ricky Shinall, William D Dutton, and Bryan A Cotton. Early prediction of massive transfusion in trauma: simple as abc (assessment of blood consumption)? *Journal of Trauma and Acute Care Surgery*, 66(2):346–352, 2009.

[40] Phillips Perera, Thomas Mailhot, David Riley, and Diku Mandavia. The rush exam: Rapid ultrasound in shock in the evaluation of the critically ill. *Emerg Med Clin North Am*, 28(1):29–56, 2010.

[41] Matthew J Pommerening, Michael D Goodman, John B Holcomb, Charles E Wade, Erin E Fox, Deborah J Del Junco, Karen J Brasel, Eileen M Bulger, Mitch J Cohen, Louis H Alarcon, et al. Clinical gestalt and the prediction of massive transfusion after trauma. *Injury*, 46(5):807–813, 2015.

[42] Gerhard Rall. Use of a pulse oxymetry sensor device, June 15 1999. US Patent 5,911,690.

[43] Donald B Rubin. Inference and missing data. *Biometrika*, 63(3):581–592, 1976.

[44] Donald B Rubin and Nathaniel Schenker. Multiple imputation for interval estimation from simple random samples with ignorable nonresponse. *Journal of the American statistical Association*, 81(394):366–374, 1986.

[45] Joseph L Schafer. *Analysis of incomplete multivariate data*. Chapman and Hall/CRC, 1997.

[46] Joseph L Schafer. *Analysis of incomplete multivariate data*. Chapman and Hall/CRC, 1997.

[47] Joseph L Schafer and John W Graham. Missing data: our view of the state of the art. *Psychological methods*, 7(2):147, 2002.

[48] George AF Seber and Alan J Lee. *Linear regression analysis*, volume 329. John Wiley & Sons, 2012.

[49] Daniel J Stekhoven. missforest: Nonparametric missing value imputation using random forest. *Astrophysics Source Code Library*, 2015.

[50] Vladimir Svetnik, Andy Liaw, Christopher Tong, J Christopher Culberson, Robert P Sheridan, and Bradley P Feuston. Random forest: a classification and regression tool for compound classification and qsar modeling. *Journal of chemical information and computer sciences*, 43(6):1947–1958, 2003.

[51] Ported to R by Alvaro A. Novo. Original by Joseph L. Schafer <jls@stat.psu.edu>. *norm: Analysis of multivariate normal datasets with missing values*, 2013. R package version 1.0-9.5.

[52] Stef Van Buuren. Multiple imputation of discrete and continuous data by fully conditional specification. *Statistical methods in medical research*, 16(3):219–242, 2007.

[53] Vladimir Naumovich Vapnik. An overview of statistical learning theory. *IEEE transactions on neural networks*, 10(5):988–999, 1999.

[54] Svante Wold, Kim Esbensen, and Paul Geladi. Principal component analysis. *Chemometrics and intelligent laboratory systems*, 2(1-3):37–52, 1987.

[55] Xianchao Xie and Xiao-Li Meng. Dissecting multiple imputation from a multi-phase inference perspective: What happens when god's, imputer's and analyst's models are uncongenial. *Statist. Sinica*, 27:1485–1545, 2017.

[56] Nedim Yücel, Rolf Lefering, Marc Maegele, Matthias Vorweg, Thorsten Tjardes, Steffen Ruchholtz, Edmund AM Neugebauer, Frank Wappler, Bertil Bouillon, Dieter Rixen, et al. Trauma associated severe hemorrhage (tash)-score: probability of mass transfusion as surrogate for life threatening hemorrhage after multiple trauma. *Journal of Trauma and Acute Care Surgery*, 60(6):1228–1237, 2006.

[57] Fuzhen Zhang. *The Schur complement and its applications*, volume 4. Springer Science & Business Media, 2006.