CHAPTER

14

Dealing with missing data, small sample sizes, and heterogeneity in machine learning studies of brain disorders

Rajat M. Thomas, Willem Bruin, Paul Zhutovsky, Guido van Wingen

Amsterdam UMC, University of Amsterdam, Department of Psychiatry, Amsterdam Neuroscience, Amsterdam, The Netherlands

14.1 Introduction

Machine learning models can be thought of as mathematical functions $\boldsymbol{F}$ that map an observation $\boldsymbol{X}$ to a corresponding discrete (classification) or continuous (regression) target $\boldsymbol{y}$. Our ability to find $\boldsymbol{F}$ hinges on the interplay between the complexity of the model and the availability of sufficient, reliable, and well-distributed data. In particular, the number of data points required to estimate a model tends to grow linearly with its complexity. Therefore, complex neural network models require millions of data points if no external regularization method is applied to force the model to favor simpler networks. In contrast, linear regression models are associated with low complexity and as such require fewer data points. Within this interplay, our ability to find the function $\boldsymbol{F}$ that generalizes to new unseen observations is affected by several factors including *missing data*, *small sample sizes*, and *heterogeneity*. The aim of this chapter is to discuss and illustrate the impact of these factors on the performance of a machine learning model.

Missing data is common issue in most real-world datasets and medical data are no exception. Missing data could be the result of different factors;

Machine Learning
https://doi.org/10.1016/B978-0-12-815739-8.00014-6

these include, for example, error in human data entry, malfunctioning of the measuring instruments, issues affecting the collection of the data (e.g., head movement in neuroimaging data), problems with the processing of the data, and, in the case of longitudinal studies, attrition of participants at follow-up. Before we talk about possible strategies to deal with this issue, it is important to mention some common sense approaches. If the missing data are confined to few features and a large proportion (>50%) of these features is missing, then it is often advisable to completely disregard them as long as there are other features available to train the machine learning algorithm. If the dataset is sufficiently large, it is also possible to use the whole dataset to train a simpler model to assess the importance of the missing features. This can help one decide whether or not to keep the missing features. Finally, if one is determined to use the features with missing data, there are several algorithms that can be employed to fill in the missing information—a procedure often termed "imputation." We discuss some of the main imputation techniques in Section 14.1.2 (Algorithms and procedures) in this chapter.

Although we are living in an era of the Big Data, there are many situations, for example, in clinical neuroimaging, where the data size is limited. Apart from a handful of large consortium studies, most of the data from patients with psychiatric or neurological disorders are collected by individual centers and analyzed locally. These datasets tend to typically comprise between 20 and 100 patients and a similar number of healthy controls (Wolfers, Buitelaar, Beckmann, Franke, & Marquand, 2015). The application of machine learning algorithms to small datasets such as these requires careful considerations. One would need to guard oneself against (i) overfitting, where the model performs well in the set of data used for training, but performs poorly in unseen data (see Chapter 2); (ii) the presence of outliers, which can positively or negatively bias the results; and (iii) the impact of noise. In Section 14.1.2, we look at some of the ways of alleviating these problems resulting from the use of machine learning with small datasets.

In neuroimaging studies of brain disorders, a possible solution to the problem of having a small dataset is *pooling*; this involves collecting data from a consortium of research and/or medical centers for the same disorder and aggregating them to generate a large dataset accessible to the members of the consortium and, in an increasing number of cases, the scientific community at large. Autism Brain Imaging Data Exchange (ABIDE),[1] Alzheimer's Disease Neuroimaging Initiative (ADNI),[2] The Enhancing Neuroimaging Genetics through Meta-analysis

[1] http://fcon_1000.projects.nitrc.org/indi/abide/.

[2] http://adni.loni.usc.edu/.

(ENIGMA)[3] Consortium, and the UK Biobank[4] are notable examples of such consortia. But the aggregation of large cohorts from various sites brings us to the final issue we deal with in this chapter: *heterogeneity*. Heterogeneity can be the result of several potential differences among sites; these include, for example, the use of different inclusion and exclusion criteria (e.g., age group, sex, medication status), different scanners (e.g., Philips, GM, or Siemens), different scanning parameters (e.g., scanning direction and voxel resolution), and different preprocessing pipelines. Although heterogeneity is often considered a nuisance, if dealt with correctly it can help one build a more robust and reliable model. The vast majority of machine learning studies in brain disorders use supervised learning to make predictions about individuals, such as group membership (e.g., patient vs. control, or remission vs. nonremission) or continuous variables (e.g., functioning or symptoms severity). In this context, heterogeneity refers to the individual variability in the relationship between input and output. For example, patients who do and do not take medication might have a different mapping from the input features, say functional connectivity, to the outcome variable of interest, for example, diagnostic category. If the dataset is small, it is important to make sure that heterogeneity is minimal and/or is corrected for within the statistical model. On the other hand, given a sufficient number of data points, the machine learning model generalizes better when the dataset has a larger variance. Therefore, heterogeneity behaves akin to model regularization by introducing "natural" variation in the data.

This chapter is organized as follows: In the next section, we elaborate a simple algorithm to generate data that are similar to a "real" dataset using pairwise correlations. This enable the reader to test the various techniques discussed later in the chapter. In the following section, we discuss the different techniques that are available to tackle the issues of *missing data*, *small sample sizes*, and *heterogeneity*. As part of this discussion, we also mention some of the state-of-the-art procedures that have yielded promising results in other areas of research but are yet to be tested in the context of brain disorders. In the final section, we provide a summary of the key recommendations.

14.2 Data simulation

To systematically test the effects of missing data, data size, and heterogeneity, we need to have detailed understanding of the properties of

[3] http://enigma.ini.usc.edu/.

[4] https://www.ukbiobank.ac.uk/.

the dataset. To enable this, we have developed a tool that allows researchers to simulate a dataset with a given number of subjects N and features f from real data. In particular, the tool generates simulated data that are similar to real neuroimaging data, with features comprising of averaged gray matter from FreeSurfer parcellations. We have made this tool available to researchers and clinicians,[5] enabling them to test the algorithms discussed in this chapter on simulated data before applying them to their own real dataset.

The procedure followed to generate the simulated data is as follows[6]:

1. Split/stratify the original dataset based on a number of criteria (**s**). In our case, we stratified based on the site of scan (~40 different sites), diagnosis (patient vs. controls), and age group (below 18 and above 18).
2. Within each of these groups **s**, calculate the correlation matrix between the features. We have chosen a dataset with 100 features (f), and therefore we compute a 100 × 100 correlation matrix, $\boldsymbol{C}$.
3. Use Cholesky decomposition to calculate a lower triangular matrix $\boldsymbol{L}$ such that $\boldsymbol{LL'} = \boldsymbol{C}$.
4. To generate a new dataset ($\boldsymbol{D}$) with N points with the same correlation structure, one needs to create an f × N matrix of random numbers ($\boldsymbol{D_{rand}}$) and multiply it with L; $\boldsymbol{D} = \boldsymbol{LD_{rand}}$.
5. **D** is an f × N matrix. Each of the features f can now be scaled by the standard deviation value of the original dataset to generate new data that are similar to the real dataset at least up to the *first moment* (based on pairwise correlations).[4] We can also offset the features by the mean of the original dataset to match the distribution of the original data.

To illustrate the validity of the approach, we show two plots. Fig. 14.1 shows the distribution of a randomly chosen feature both for the simulated and the real dataset. There is considerable overlap in the distribution, which indicates the data have been scaled appropriately. Fig. 14.2 shows the linear associations between the variables as a heatmap of the correlation matrix. It can be seen that the simulated data have been able to capture the correlation structure on the larger scale.

The usefulness of creating a dataset similar to the one to be used for a machine learning problem is the ability to test the impact of different imputation and machine learning approaches on simulated dataset, thereby reducing the risk of overfitting on the actual dataset of interest,

[5] Gitlab repo: https://github.com/rajatthomas/data_preparation contains all the code associated with the dataset generating process described below.

[6] This is an example based on our real-world dataset. Readers can adapt these steps based on their own dataset.

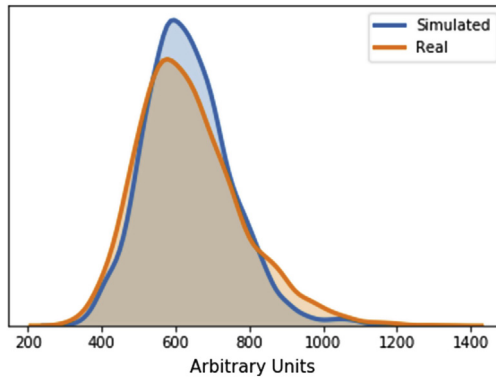

FIGURE 14.1 Distribution of a real versus a simulated brain volume feature.

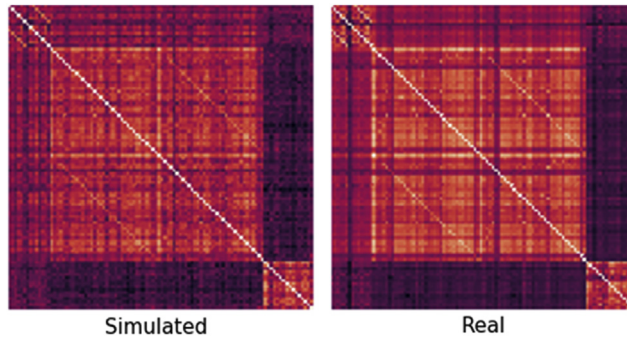

FIGURE 14.2 Heatmaps depicting the correlation matrices of both real and simulated data.

i.e., handcrafting a model that works well only for the original dataset. A caveat to bear in mind, however, is that the generalization of the model might rely on the fact that the out-of-sample data (test data) were generated using the same mechanism as the training data, which may not be true in all cases.

14.3 Algorithms and procedures

14.3.1 Missing data

As alluded to before, procedures that are aimed at *guessing* the missing value in a dataset are referred to as imputation methods. Imputation tries to fill in the values of missing data while preserving the characteristics of

their distribution and maintaining the relationships to other variables. Imputations can provide huge benefits in the context of clinical trials, for example, by reducing the bias associated with the inclusion of patient with complete data and maximizing the use of the available information (Burton, Billingham, & Bryan, 2007). There have been several studies looking at the impact of imputation both for continuous (Burton et al., 2007, Ghosh-Dastidar & Schafer, 2003) and categorical missing data (Eisemann, Waldmann, & Katalinic, 2011).

All algorithms used for imputation make assumptions regarding the distribution of the missing data. Therefore, before choosing an algorithm for imputation, it is crucial to find out the pattern and causes of the missing data; in other words, which observations are missing and why? Following van Buuren (2012), we can categorize the pattern of missing data as (i) univariate, i.e., missingness is confined to one feature variable, (ii) monotone, i.e., all values are missing in a feature after a specific point (e.g., no values beyond a certain date), (iii) connected, i.e., the missing value for a feature co-occurs with the missing value for another feature (e.g., date of birth and age are either both present or both absent), or, alternatively, two missing values co-occur within the same feature (e.g., number of hours worked is missing every weekend), and (iv) random, i.e., values are missing with no apparent pattern.

Having described the taxonomy of the missing patterns, next we define the possible mechanisms that can generate these missing data. Theoretically, every data point has a likelihood of being *missing*. The underlying mechanism that governs this probability is called *missing data mechanism* or *response mechanism*. The model that describes this mechanism is called *missing data model* or *response model*. Little and Rubin (2014) introduced a nomenclature (more details also available in Rubin, 1976) for these missing data mechanisms: missing completely at random (MCAR), missing at random (MAR), and missing not at random (MNAR).

When the probability of a data point missing is the same in all cases (e.g., in all diagnostic groups) throughout the dataset, the data are said to be MCAR. Here, the underlying assumption is that, no matter which group the features belong to, the fact that a feature is missing is equally likely across the groups. While this is a very convenient assumption and the easiest to deal with, it is often not realistic. For example, in an investigation using machine learning to predict clinical response to a certain treatment of interest, the likelihood of a missing measurement in a group of patients shows that good response might be higher than in a group of patients who show poor response.

A more general case that is more applicable to real-world contexts is the MAR case. This time we do not assume that this probability is the same *across* all groups (e.g., responders and nonresponders); instead we only assume that the probability of missingness is the same *within* a group

(e.g., among responders). While more applicable, this assumption can also be unrealistic in several real-life situations. For example, if the goal is to investigate the difference between responders and nonresponders to a certain treatment, one can imagine that even within the responders group those who show higher levels of functioning are more likely to show up for a longitudinal follow-up than those who show lower level of functioning. This will cause a systematic bias in the missingness. This last point brings us to the MNAR case, which is the most difficult case for imputation, as the missing data come from a specific distribution selectively, i.e., the probability of a data point being missing is not random, for example, the case when, within each group, data from patients with low level of functioning are more likely to be missing than data from patients with high levels of functioning.

We should therefore think carefully about the mechanism underlying missing data before applying a specific algorithm. The most simple algorithms work well only in the case of the MCAR assumption and will produce biased results in other cases. Below, we discuss some of the solutions to impute data under different scenarios.

1. *Complete case analysis/listwise deletion*: By far the simplest and the most wasteful technique, listwise deletion involves the elimination of every row/column having missing data. Under the MCAR assumption, this technique produces unbiased estimates of means, variances, and regression weights (van Buuren, 2012) and provides the right standard errors and statistical significance testing for the reduced dataset. The biggest drawback of this technique is that, in most cases, we lose over 50% of the dataset (Little & Rubin, 2014). Also, if the MCAR assumption is not valid, deleting missing data might have the unwanted effect of excluding the majority of the entire population or subtypes. Therefore, the only times it is justified to apply listwise deletion is when there are only a few missing values compared to the size of the dataset. As an illustrative example (Fig. 14.3), we simulated a dataset with 4000 data points, each having 10, 20, 100, or 200 features. From this dataset, we eliminated between 0% and 5% of the entries randomly. Then, we implemented the linewise deletion procedure which removes every row with at least one feature that is missing. As can be seen from Fig. 14.3, by the time we reach about 3% of missing data, we have removed the entirety of the data.
2. *Pairwise deletion*: If a machine learning algorithm (e.g., regression) only relies on means of features and pairwise correlations between features, then the pairwise deletion approach can be applied. The idea is simple: calculate means of each feature only using the nonmissing data and correlations between features only using the

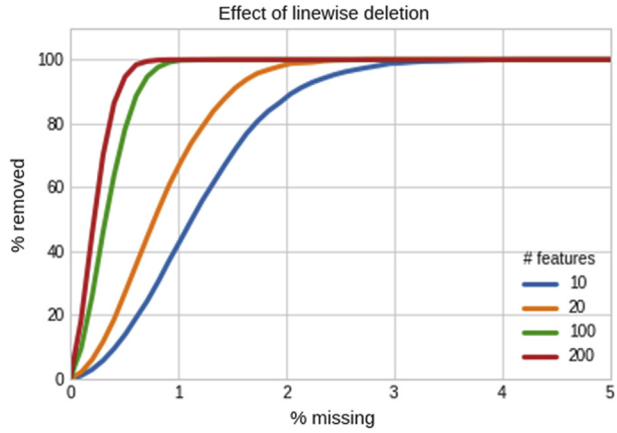

FIGURE 14.3 Catastrophic data loss with linewise deletion.

subset of the two features with no missing data. A key advantage of this approach is that it allows the use of all data; however, calculating errors on the estimates can be difficult (Marsh, 1998).

3. *Mean/median/mode imputation*: This is one of the most used imputation techniques. This approach is univariate in the sense that it only deals with one feature at a time. Before deciding to do either mean or median, one has to plot the distribution of the feature (using the nonmissing data). If the distribution is approximately Gaussian, mean imputation can be used, whereas if the distribution has a long tail (skewed distribution), median imputation should be preferred. For a few missing values per feature, this technique often gives good results (Schmitt, Mandel, & Guedj, 2015). For categorical variables, the mean is replaced by the mode of the distribution.
4. *Intermediate-level approaches*: Apart from the simple approaches mentioned above, other commonly used techniques include k-nearest neighbors (kNN), singular value decomposition (Troyanskaya et al., 2001), fuzzy k-means (Li, Deogun, Spaulding, & Shuart, 2004), Bayesian principal component analysis (Oba et al., 2003), and multiple imputations by chained equations (Rubin, 2004). The key element that differentiates these methods from the others is that they are all multivariate methods of imputation. This means that imputation is performed considering all features as a single vector. Vectors with missing values are therefore "matched" to the closest set of vectors for which all data are available. For example, in the case of the kNN, the kNN are found using the features for which the data are available. The missing feature is then imputed as the weighted (inversely to the distance to the point) mean of that feature in the kNN. Schmitt et al. (2015) performed a comparison between

simpler and more advanced techniques. The results indicated that there is no silver bullet and that it is advisable to try a few techniques (e.g., using simulations) before determining which one to use.

5. *Deep learning approaches*: In a recent study (Yoon, Jordon, & van der Schaar, 2018), missing data are imputed by adapting the well-known generative adversarial networks (GANs) framework. This method, known as the generative adversarial imputation networks (GAINs), uses a generator (G) that observes some components of a real data vector and imputes the missing components conditioned on the nonmissing data and outputs an imputed vector. The discriminator (D) then takes this complete vector and attempts to determine which components were actually observed and which were imputed. GAIN seems to outperform several other imputation techniques.

In short, imputation is a vast topic dating back decades. Here, we have given the reader an overview of popular techniques, together with some rules of thumb on how to use them. Finally, it is helpful to mention an important precaution that needs bearing in mind when it comes to using imputed values for machine learning. The imputation must be done on the training and test dataset separately. One should not use the whole dataset to impute and then subdivide it into training and test datasets. This is a common pitfall resulting in information leakage in machine learning.

14.3.2 Small sample sizes

As mentioned in the introduction, small sample sizes are a common issue in the investigation of brain disorders. An additional complication specific to neuroimaging data is the large difference between the number of features and the amount of data available: magnetic resonance imaging (MRI) images can comprise more than 100,000 voxels—each of which can be considered as a feature—while the typical number of participants tends to be much smaller (e.g., up to 100). In this case, any model fit to the data will be highly undetermined (i.e., there are different possible model parameters that can explain the data rather than a single parameter), and one needs to employ methods to deal with the large imbalance between number of feature and number of subjects. Before discussing some of these methods, we mention a few practical strategies for maximizing the use of small datasets.

1. *A targeted question* as opposed to "data exploration": Rather than trying to find associations between say resting-state functional activation and the presence/absence of autism, which entails a plethora of various possible techniques, the question should be

refined, for example, "does the correlation matrix based on a priori regions of interest predict autism?"

2. *Simple models*: Restricting the hypothesis space of plausible models will give one a better chance of finding a good solution. For example, one can select linear models with no feature interaction terms (unless there is strong reason to believe that such interaction exists). If one is using decision trees, it is also possible to restrict the hypothesis space by limiting the depth of the tree. *Regularization* (such as L1/L2) can be a useful technique to search for a simple model in a data-driven fashion; specifically, strong regularization can help one fit a large model to a small dataset.
3. *Feature selection/transformation*: Another approach for maximizing the use of small datasets in machine learning involves preselecting the features to be used. Here, it is important to note that preselection of this sort should not be based on the data that one is analyzing. Instead, preselection should be based on prior knowledge based on independent datasets (for example, a review of the existing literature). Feature transformation is powerful, especially in small data. For example, let us imagine a situation in which one is investigating the relationship between structural neuroanatomy and severity of clinical depression using structural MRI. A useful way to transform and select features could be to transform the whole-brain, voxel-wise data into brain regions and select the most relevant one for depression. Feature selection does not have to be based on prior knowledge; instead it can also be performed automatically using data from the training set. There are three categories of feature selection approaches: (1) Filter methods: univariate techniques which try to select the features that are most closely associated with the target variable. An example of such methods would be a correlation between each feature and the target value, subsequently followed by the selection of only the most highly correlated features to be considered for training the machine learning model. (2) Wrapper methods: multivariate techniques which train a machine learning model on different subset of features, searching for the combination of features which provides optimal performance. Examples include forward/backward feature selection and recursive feature elimination. (3) Embedded methods: in this case the feature selection process and model training are combined. An example is L1 regularization of a regression model which will automatically determine the most relevant subset of features.
4. *Clean data*: One of the key problems with small datasets is the low signal-to-noise ratio, which affect the ability to perform a machine learning task. To address this issue a number of transformations can

be applied to the data to maximize the signal-to-noise ratio. In the context of neuroimaging, for example, preprocessing steps such as realignment, slice time correction, and registration might help the machine learning algorithm hone in on the relevant features more easily.

5. *Model averaging*: When performing machine learning with small datasets, different algorithms may provide the best fit to the distribution of the data in different regions. An ensemble model, combining the predictions from several models, could prove beneficial for maximizing the use of the available data.
6. *Using confidence intervals rather than point estimates*: It is recommended to find ways of defining a confidence interval (resampling, bootstrapping, dropouts) rather than simply using point estimates, especially in the context of small datasets.

Having mentioned the main practical strategies for maximizing the use of small datasets, we now discuss four methods that have been shown to be useful:

1. *Data augmentation*: A simple technique in which the input data are replicated by translating, rotating, or distorting it. This assumes that these transformations to the data do not alter the target label. UNET (Ronneberger, Fischer, & Brox, 2015) first popularized this technique in medical imaging for the segmentation of histopathology; in particular, the researchers made use of image distortion for augmentation which proved effective. Similar image distortion can also be applied to neuroimaging data, for example, using the DLTK toolkit (Pawlowski et al., 2017).
2. *Transfer learning*: A machine learning technique where models trained in one domain can be reused, either as they are or with minor fine-tuning, for tasks performed in another domain (Canziani, Paszke, & Culurciello, 2016). For example, in a recent paper (Ghafoorian et al., 2017), transfer learning was used to improve brain lesion segmentation. There is an increasing number of large-scale neuroimaging datasets available to the research community (e.g., UK Biobank). This provides researchers with the opportunity to train a model (for example, a convolutional neural network) to perform a task using a large-scale dataset and then fine-tune the last few layers of the network using their own smaller dataset, in effect transferring the learning from existing datasets to their own.
3. *Simulation-based augmentation*: GANs (Goodfellow et al., 2014) are a family of machine learning architectures that can generate new images from a particular domain. They are implemented as two neural networks competing against each other in a zero-sum game. Shin et al. (2018) successfully used this technique to create additional

brain images with a tumor. This enabled them to not only increase their sample size, which in turn benefited the performance of the model, but also to ensure anonymization of the data. Frid-Adar et al. (2018) used a similar expedient to increase classification of liver lesions from computed tomography (CT) images.

4. *Data efficient learning*: State-of-the-art algorithms exploit symmetries in the input data by rethinking the design of convolutional layers. One such technique is the so-called group convolution network (Cohen & Welling, 2016). This technique has shown promise in pulmonary node detection in lung CT scans (Winkels & Cohen, 2018) and could easily extended to neuroimaging datasets.

Until recently, large databases were pivotal to the success of machine learning (in particular, deep learning models which are especially "data hungry"). But, with the increasing availability of strategies and techniques such as those discussed above, we are entering an era where complex models can be used effectively with smaller datasets.

14.3.3 Heterogeneity

Heterogeneity, in this context, refers to the inherent variability in the underlying characteristics of a population from which data are sampled. Heterogeneity broadly can be of two types: (i) systematic and (ii) "by-chance." There are a number of variables such as age, sex, and medication status that can have an influence on the outcome of a study. But these heterogeneities are known before and can be controlled for—to a certain extent—for example, by constraining the age, sex, and/or medication status of participants to ensure the final sample is homogeneous with respect to these variables. Even when the inclusion and exclusion criteria are very stringent, however, there are some types of heterogeneity that cannot be avoided. For example, it is hypothetically possible that the same investigation might reveal different outcomes when performed at different times, due to slight differences in the sensitivity of the tool used to collect the data. Such differences may be impossible to predict and correct but, in the context of large studies, it is hoped that they will be randomly distributed and therefore will have minimal or no impact on the statistical comparisons of interest.

Studies that are smaller in size are often conducted in a controlled fashion to minimize the effects of systematic heterogeneity. Large-scale multisite studies, on the other hand, tend to be inherently heterogeneous because of the requirement to use broader inclusion and exclusion criteria that can be applied across sites. Smaller studies thus focus on answering a specific question about their patient population, whereas larger studies assume that a fundamental pattern of the disorder of

interest can be detected despite the presence of heterogeneities. In other words, small and large studies are geared toward answering complementary questions about a particular disorder (Nunes et al., 2018). Therefore, when performing a machine learning task, one is likely to obtain higher accuracies in small homogeneous studies at the cost of poor generalizability and good generalizability in large heterogeneous studies at the cost of lower accuracy (Schnack & Kahn, 2016). A number of recent reviews (Kambeitz et al., 2015; Zarogianni, Moorhead, & Lawrie, 2013) discuss the critical trade-off between homogeneous small sample and heterogeneous large sample.

In the remaining part of this section, we consider the main procedures for dealing with data heterogeneity:

1. *Strict selection criteria*: An obvious way to avoid systematic heterogeneity is to use strict selection criteria that will minimize heterogeneity in the data. This can be done, for example, by including only male or female participants, restricting the age range, using strict clinical criteria, and so on. Although an easy fix, this procedure tends to be used in single-site studies, where it is more practical to adopt specific demographic and clinical criteria. Another caveat with this procedure is that the results are unlikely to extend to the general population due to the use of a selected sample based on strict criteria.
2. *Exploration of the effects*: The systematic (known) causes of heterogeneity can be modeled as confounds in a machine learning task. To get an idea of the effect of these confounds, it is recommended to perform an exploration of their effects. For example, one can start by performing a classification task only based on the confounding variables (e.g., which site the data came from). If the accuracy hovers around chance level, one can conclude that there is no direct effect of the confounding variables on the output, although there still could be an interaction effect with the variable or features of interest. As another test, one can correlate all confounding variables with the features of interest: if there is a significant correlation, this can be addressed using one of the following techniques.
3. *Regress out the confounds*: If a statistical analysis is restricted to general linear models, one can include confounding variables as additional *features of no interest.* Looking at a measure of the variance explained will elucidate the effect of these variables on the data. A popular step is also to regress out the effect of the confounds from the variables of interest *before* the actual machine learning is performed. This can be done in an iterative scheme by orthonormalizing each of the variables of interest in turn with every

confounding variable. But in a multisite study, this may not always be desirable because the effect of the confounds (say gender, age, or medication status) on a feature (for example, gray matter volume) may have a linear relationship but the slope could change from one site to another (for example, due to an interaction with the scanner). In this situation, if a particular site has sufficient cases, it is advisable to regress out confounds at the site level (Snoek, Miletić, & Scholte, 2019). It is good to emphasize that this should be done only on the training split of the dataset and then apply the coefficients to the test split.

4. *Access the effect of the confounds*: A recent study (Nunes et al., 2018) measured the effect of the confounds, including demographic and clinical variables, on the results of a classification task using mixed-effects logistic regression. The clinical variables were treated as random effects on top of group averages per site. A point to note here is that the classification was carried out without using the confounding variables as features. This type of analysis can help clarify whether the main effect found in the classification is due to the underlying heterogeneity rather than the variable of interest.
5. *Data harmonization*: As mentioned before, site-related differences (for example, in recruitment criteria) can make the combination of different datasets challenging. In a recent multisite neuroimaging study (Rozycki et al., 2018), the authors were able to identify a robust signature of schizophrenia by "harmonizing" the data across sites. Their procedure involved estimating the effects of intracranial volume, site, age, and sex effects on each feature within a *pooled sample of controls* using a linear model and then applying the coefficients estimated from this model to the whole sample including patient data. Effectively, this removed the influence of site and demographic effects on the difference between patients and controls. Importantly, this control-based harmonization model was always cross-validated; in other words, it was estimated using the training set and subsequently applied to the test set. The harmonization procedure was applied independently on each feature.

14.4 Conclusions

In this chapter, we have defined the issues of *missing data*, *small sample sizes*, and *heterogeneity* in machine learning datasets and discussed possible solutions using examples from neuroimaging. We have explored classical techniques employed in the literature to tackle these issues as

well as state-of-the-art approaches that use neural networks and their variants. We have also introduced a simple tool to generate data that are similar to an existing dataset; we hope the tool will enable readers to test the various techniques while minimizing the risk of overfitting the actual data.

The first problem we discussed is that of missing data and how one might fix it using imputation methods. The success of an imputation method depends on the extent to which one is able to estimate the underlying distribution of the data. Because there is no imputation method that performs best at all times, it is recommended to test the main imputation scheme either on simulated data or a subset of the data. Methods that take into account all the features (multivariate) seem to fare better than univariate methods which consider a single feature at the time. Finally, if there are only few missing points, it might be advisable to disregard those data points instead of making use of imputation.

Apart from noisy or missing data, a defining feature of most clinical studies is the small amount of available data. Controlling for model complexity (either by increased regularization or the use of fewer parameters) and trading point estimates for confidence intervals are some strategies to bear in mind while dealing with smaller datasets. Advanced techniques such as transfer learning make it possible to take pretrained models that have been trained on large datasets and fine-tune them to smaller datasets. In short, the core principles to follow when dealing with small datasets are to define a specific question and perform robust preprocessing and feature selection that are informed by good domain knowledge.

Finally, when going from small to big data, one of the most challenging issues is that of heterogeneity. Heterogeneity is often a double-edged sword: ideally, one wants the maximal variation in the variables of interest (including dependent and independent variables) and minimal variation in variables of no interest (confounds). Data exploration to identify the relationship between variables of interest and confounds is pivotal before the application of one of the techniques mentioned. If handled correctly, heterogeneity can be used to one's advantage to build more robust models and discover true underlying patterns in brain disorders.

It is also important to reiterate that, when trying out different algorithms (for example, different imputation schemes), one must do so using separate training and testing datasets. Imputation using the entire dataset is a common cause of information leakage in machine learning datasets. A good understanding of the underlying data and their underlying distribution can go a long way in determining the right set of tools to use; for example, exploring a feature, its distribution, and its relationship to other variables including confounds. Perhaps the most important lesson when

exploring these problems is that there is not one particular pipeline that can be applied to all problems. Instead, exploratory data analysis is needed to pin down the right steps for a certain task. What we have provided here is an overview of the main strategies and techniques on how to go about this "unglamorous" but critical aspect of applying machine learning to brain disorders.

14.5 Key points

- *Missing data* can result from error in human data entry, malfunctioning of the sensors/instruments, software bugs within the acquisition, and preprocessing pipelines or patient attrition.
- *Small sample sizes* are the norm when it comes to studies of clinical populations; this contrasts with the large amount of data required by advanced machine learning algorithms.
- *Heterogeneity*, although inherent in all studies small and large, is especially prominent in the case of multisite studies.
- To deal with missing data, exploring the *data (feature) distributions* and interdependence (*feature cross-correlation*) is pivotal in choosing the right imputation scheme, i.e., data completion strategy.
- Simple questions, data cleaning, feature selection, and simple models are all good nonalgorithmic rules of thumb to bear in mind when dealing with small sample sizes.
- The impact of heterogeneity can be minimized by harmonizing the data with respect to possible confounds before estimating the machine learning model; or it can be assessed after estimating the model.
- *One size does not fit all* when it comes to using the algorithms discussed. Perform an exploratory data analysis to determine the right course of action.

Acknowledgments

This work was supported by the Netherlands Organization for Scientific Research (NWO/ZonMW Vidi 016.156.318).

References

Burton, A., Billingham, L. J., & Bryan, S. (2007). Cost-effectiveness in clinical trials: Using multiple imputation to deal with incomplete cost data. *Clinical Trials, 4*(2), 154–161.

van Buuren, S. (2012). *Flexible imputation of missing data*. CRC Press.

Canziani, A., Paszke, A., & Culurciello, E. (2016). *An analysis of deep neural network models for practical applications*. arXiv preprint arXiv:1605.07678.

Cohen, T., & Welling, M. (2016). June). Group equivariant convolutional networks. In *International conference on machine learning* (pp. 2990–2999).

Eisemann, N., Waldmann, A., & Katalinic, A. (2011). Imputation of missing values of tumour stage in population-based cancer registration. *BMC Medical Research Methodology, 11*(1), 129.

Frid-Adar, M., Diamant, I., Klang, E., Amitai, M., Goldberger, J., & Greenspan, H. (2018). *GAN-based synthetic medical image augmentation for increased CNN performance in liver lesion classification*. arXiv preprint arXiv:1803.01229.

Ghafoorian, M., Mehrtash, A., Kapur, T., Karssemeijer, N., Marchiori, E., Pesteie, M., et al. (September 2017). Transfer learning for domain adaptation in mri: Application in brain lesion segmentation. In *International conference on medical image computing and computer-assisted intervention* (pp. 516–524). Cham: Springer.

Ghosh-Dastidar, B., & Schafer, J. L. (2003). Multiple edit/multiple imputation for multivariate continuous data. *Journal of the American Statistical Association, 98*(464), 807–817.

Goodfellow, I., Pouget-Abadie, J., Mirza, M., Xu, B., Warde-Farley, D., Ozair, S., et al. (2014). Generative adversarial nets. In *Advances in neural information processing systems* (pp. 2672–2680).

Kambeitz, J., Kambeitz-Ilankovic, L., Leucht, S., Wood, S., Davatzikos, C., Malchow, B., et al. (2015). Detecting neuroimaging biomarkers for schizophrenia: A meta-analysis of multivariate pattern recognition studies. *Neuropsychopharmacology, 40*(7), 1742.

Li, D., Deogun, J., Spaulding, W., & Shuart, B. (June, 2004). Towards missing data imputation: A study of fuzzy k-means clustering method. In *International conference on rough sets and current trends in computing* (pp. 573–579). Berlin, Heidelberg: Springer.

Little, R. J., & Rubin, D. B. (2014). *Statistical analysis with missing data* (Vol. 333). John Wiley & Sons.

Marsh, H. W. (1998). Pairwise deletion for missing data in structural equation models: Nonpositive definite matrices, parameter estimates, goodness of fit, and adjusted sample sizes. *Structural Equation Modeling: A Multidisciplinary Journal, 5*(1), 22–36.

Nunes, A., Schnack, H. G., Ching, C. R., Agartz, I., Akudjedu, T. N., Alda, M., et al. (2018). Using structural MRI to identify bipolar disorders–13 site machine learning study in 3020 individuals from the ENIGMA bipolar disorders working group. *Molecular Psychiatry, 1*.

Oba, S., Sato, M. A., Takemasa, I., Monden, M., Matsubara, K. I., & Ishii, S. (2003). A Bayesian missing value estimation method for gene expression profile data. *Bioinformatics, 19*(16), 2088–2096.

Pawlowski, N., Ktena, S. I., Lee, M. C., Kainz, B., Rueckert, D., Glocker, B., et al. (2017). *Dltk: State of the art reference implementations for deep learning on medical images*. arXiv preprint arXiv:1711.06853.

Ronneberger, O., Fischer, P., & Brox, T. (October, 2015). U-net: Convolutional networks for biomedical image segmentation. In *International Conference on Medical image computing and computer-assisted intervention* (pp. 234–241). Cham: Springer.

Rozycki, M., Satterthwaite, T. D., Koutsouleris, N., Erus, G., Doshi, J., Wolf, D. H., et al. (2018). Multisite machine learning analysis provides a robust structural imaging signature of schizophrenia detectable across diverse patient populations and within individuals. *Schizophrenia Bulletin, 44*(5), 1035–1044.

Rubin, D. B. (1976). Inference and missing data. *Biometrika, 63*(3), 581–592.

Rubin, D. B. (2004). *Multiple imputation for nonresponse in surveys* (Vol. 81). John Wiley & Sons.

Schmitt, P., Mandel, J., & Guedj, M. (2015). A comparison of six methods for missing data imputation. *Journal of Biometrics and Biostatistics, 6*, 1.

Schnack, H. G., & Kahn, R. S. (2016). Detecting neuroimaging biomarkers for psychiatric disorders: Sample size matters. *Frontiers in Psychiatry, 7*, 50.

Shin, H. C., Tenenholtz, N. A., Rogers, J. K., Schwarz, C. G., Senjem, M. L., Gunter, J. L., et al. (September 2018). Medical image synthesis for data augmentation and anonymization using generative adversarial networks. In *International workshop on simulation and synthesis in medical imaging* (pp. 1–11). Cham: Springer.

Snoek, L., Miletić, S., & Scholte, H. S. (2019). How to control for confounds in decoding analyses of neuroimaging data. *NeuroImage, 184*, 741–760.

Troyanskaya, O., Cantor, M., Sherlock, G., Brown, P., Hastie, T., Tibshirani, R., et al. (2001). Missing value estimation methods for DNA microarrays. *Bioinformatics, 17*(6), 520–525.

Winkels, M., & Cohen, T. S. (2018). *3D G-CNNs for pulmonary nodule detection*. arXiv preprint arXiv:1804.04656.

Wolfers, T., Buitelaar, J. K., Beckmann, C. F., Franke, B., & Marquand, A. F. (2015). From estimating activation locality to predicting disorder: A review of pattern recognition for neuroimaging-based psychiatric diagnostics. *Neuroscience and Biobehavioral Reviews, 57*, 328–349.

Yoon, J., Jordon, J., & van der Schaar, M. (2018). *Gain: Missing data imputation using generative adversarial Nets*. arXiv preprint arXiv:1806.02920.

Zarogianni, E., Moorhead, T. W., & Lawrie, S. M. (2013). Towards the identification of imaging biomarkers in schizophrenia, using multivariate pattern classification at a single-subject level. *NeuroImage: Clinical, 3*, 279–289.