Missing Data and Uncertainty in Batch Reinforcement Learning

Daniel J. Lizotte **Lacey Gunter** **Eric Laber** **Susan A. Murphy**

{danjl,lgunter,laber,samurphy}@umich.edu
Department of Statistics, University of Michigan, Ann Arbor, MI 48109, USA

Abstract

We present a method for batch Q-learning when some of the data are missing. Our approach uses Bayesian multiple imputation to build Q-functions using all of the observed data. This is safer than using complete case analysis, i.e. throwing out incomplete training samples, because it can avoid non-response bias when possible. We also present a method for assessing confidence in the learned Q-functions, and demonstrate our methods on real multistage clinical trial data.

1 Introduction

In the batch, finite-horizon reinforcement learning (RL) setting, we estimate the optimal action value function $Q^*(o, a)$ using an estimator $\hat{Q}(o, a; \mathcal{Y})$ of the expected sum of future rewards one would obtain by taking action a given observation o and acting optimally thereafter. This estimator is constructed using a complete dataset $\mathcal{Y}$, which is a set of training trajectories where all of the observation, action, and reward variables within each trajectory are known. Various regression models could be be used to construct $\hat{Q}$ depending on the particular application we have in mind, e.g. lookup tables, linear regression, neural networks, etc. In the finite-horizon setting, $\hat{Q}$ can be computed using one backward pass of dynamic programming starting with the final observations, actions and rewards, and computing maximizations over estimated expected rewards as we move backward toward the beginning of the trajectories [7]. In this setting, reinforcement learning (Q-learning) is achieved by repeated application of supervised learning (regression.)

We are interested in the case where some of the elements of the sample trajectories in $\mathcal{Y}$ are missing. This missingness could involve any of the observations, actions, or rewards in the trajectories in $\mathcal{Y}$, and could be different from trajectory to trajectory. We write $\mathcal{Y} = (\mathcal{Y}_{\text{obs}}, \mathcal{Y}_{\text{mis}})$ to denote the portions of the data $\mathcal{Y}$ that are observed and missing, respectively. If some data are missing, we can construct an estimator $\hat{Q}_{\text{obs}}$ derived from $\hat{Q}$ that depends only on the observed data $\mathcal{Y}_{\text{obs}}$. We do this by taking an expectation over the missing data $\mathcal{Y}_{\text{mis}}$:

$$\hat{Q}_{\text{obs}}(o, a; \mathcal{Y}_{\text{obs}}) = \mathbb{E}_{\mathcal{Y}_{\text{mis}}|\mathcal{Y}_{\text{obs}}}[\hat{Q}(o, a; \mathcal{Y})] = \int \hat{Q}(o, a; \mathcal{Y}) P(\mathcal{Y}_{\text{mis}}|\mathcal{Y}_{\text{obs}}) \mathrm{d}\mathcal{Y}_{\text{mis}} \; . \tag{1}$$

In this work, we describe the settings where this approach is sensible and important, followed by a model for $P(\mathcal{Y}_{\text{mis}}|\mathcal{Y}_{\text{obs}})$ that is commonly used when working with partially missing data, and a method for obtaining confidence measures of $\hat{Q}_{\text{obs}}$. We then present a case study where we use this approach to analyze real multi-stage clinical trial data.

2 Random Missingness and Ignorability

Most supervised learning algorithms presume we have a collection of completely observed, labeled, i.i.d. training samples (analogous to “training trajectories” in the batch RL setting) which we use to estimate some function over the input (feature) space. If within a training sample some feature values are unknown or the label is unknown, a decision must be made about how that partially-observed training sample should influence the output of the learning algorithm. Many popular models, e.g. linear regression, rely on complete observation of the features and labels in each training sample.

One naïve but commonly-used approach to using an algorithm designed for complete data on a dataset where some values are missing is to simply ignore any training samples that are not completely observed. This is known as "complete case analysis" (CCA). The appropriateness of CCA depends on the *missingness mechanism* of the data. Suppose the probability that the value of Y_i is missing does not depend in any way on the values of $\mathcal{Y}_{\text{mis}}$ or $\mathcal{Y}_{\text{obs}}$, but only on a parameter Φ. For example, imagine that for each observable value, we flip a coin where $P(\text{heads}) = \Phi$, and hide the value if the coin turns up 'heads.' This type of data is termed *missing completely at random* (MCAR.) If the data $\mathcal{Y}$ are MCAR and we construct a dataset $\mathcal{Y}^{\text{CCA}}$ by deleting the incomplete rows of $\mathcal{Y}$, this dataset will have same distribution as a dataset $\mathcal{Y}^{\text{nomis}}$ of the same size that has rows drawn from the same distribution as $\mathcal{Y}$ but *without* the missingness mechanism applied. Intuitively, in the MCAR case, throwing out the incomplete training samples will not have any effect on a supervised learner beyond reducing the size of the training set: its expected bias, variance, and other properties will remain the same. Of course, throwing out incomplete training samples may also throw out large amounts of information about quantities we are trying to predict.

More complex mechanisms of missingness are common in practice. If the probability that a value is missing depends on Φ and also on $\mathcal{Y}_{\text{obs}}$, but does not depend on the hidden values in $\mathcal{Y}_{\text{mis}}$, then the data are termed *missing at random* (MAR.) This covers a wider class of missingness mechanisms that includes MCAR.

Suppose for example that we are monitoring a group of patients with a chronic disease. We administer an inexpensive test to all of the patients that assesses their symptom severity, categorizing each patient's symptoms as "high severity" or "low severity." We also have a second more expensive test that gathers more detailed biological information. We administer this second test to 90% of the "high severity" patients, and to 50% of the "low severity" patients. In this case, applying CCA, i.e. deleting all people with a missing test result, will leave a population that has had "low severity" patients differentially deleted, since more of the training samples for that subgroup will be incomplete. The remaining population will almost certainly have a different distribution from the original random sample of patients.

Using CCA for data that are MAR is clearly inappropriate. However, some methods that use all of $\mathcal{Y} = (\mathcal{Y}_{\text{mis}}, \mathcal{Y}_{\text{obs}})$ can safely be used with MAR data for inference, even if they do not model the missingness mechanism explicitly. In particular, Bayesian inference that assumes a generative model of $\mathcal{Y}$ given parameters Θ and supposes $\mathcal{Y}_{\text{obs}} \sim \int P_\Theta(\mathcal{Y}_{\text{obs}}, \mathcal{Y}_{\text{mis}}) \mathrm{d}\mathcal{Y}_{\text{mis}}$ effectively ignores the missingness mechanism; however, this assumption yields correct posterior distributions when the data are MAR and the missingness is "ignorable," and this procedure is termed a "general ignorable procedure."[1]

There are many cases where we cannot be certain that data are completely MAR, nor that missingness is completely ignorable. However, there is strong empirical evidence [6] that even when these assumptions do not hold, the better we can model the missing data given the observed data, the more we will correct for "non-response bias"—the type of bias induced by applying CCA. This in turn is strong evidence for reasoning about missing data using a model that describes the entire dataset $\mathcal{Y}$, and for making that model expressive enough to capture the relationships between the observed and the missing data. Even if the probability of missingness depends on the values of unobserved quantities, the more accurately we can predict those quantities the less bias we will incur.

All of these concepts related to missing data were developed in detail by Rubin [4], and are discussed in the context of the models we will present in this paper by Schafer [6] and others.

3 A Model for Missing Data: Bayesian Multiple Imputation

In general, we are not willing to make the MCAR assumption that would allow us to use complete case analysis to "solve" our missing data problem, nor are we willing to throw out potentially valu-

[1] The ignorability assumption means that a priori, Θ and Φ are independent. For details, see Schafer [6].

able data even if the missingness is MCAR. We therefore use a general ignorable procedure for constructing a function $\hat{Q}_{\text{obs}}$ to estimate Q^* using all of the observed data:

$$\begin{aligned}\hat{Q}_{\text{obs}}(o, a; \mathcal{Y}_{\text{obs}}) &= \mathbb{E}_{(\mathcal{Y}_{\text{mis}},\Theta)|\mathcal{Y}_{\text{obs}}}[\hat{Q}(o, a; \mathcal{Y})] && (2)\\ &= \iint \hat{Q}(o, a; \mathcal{Y}) P(\mathcal{Y}_{\text{mis}}|\mathcal{Y}_{\text{obs}}, \Theta) P(\Theta|\mathcal{Y}_{\text{obs}}) \mathrm{d}\mathcal{Y}_{\text{mis}} \mathrm{d}\Theta \, . && (3)\end{aligned}$$

The function $\hat{Q}_{\text{obs}}$ is the posterior mean of the $\hat{Q}$ functions given the observed data $\mathcal{Y}_{\text{obs}}$ and the model parameters Θ. Other estimates are possible, particularly in this Bayesian setting; one could use the posterior mode of $\hat{Q}$, for example. The usefulness of other estimators is an area for future investigation.

The integral (3) over all possible values of $\mathcal{Y}_{\text{mis}}$ and Θ is intractable for many interesting models, including the one we present in the next section. We will therefore obtain samples of $\mathcal{Y}_{\text{mis}}|\mathcal{Y}_{\text{obs}}$ from which we construct multiple imputed data sets $\mathcal{Y}^{(1)}, \mathcal{Y}^{(2)}, ..., \mathcal{Y}^{(m)}$, which in turn allow us to approximate (3) using an empirical average

$$\hat{Q}_{\text{obs}}(o, a; \mathcal{Y}_{\text{obs}}) \approx \frac{1}{m} \sum_i \hat{Q}(o, a; \mathcal{Y}^{(i)}) \qquad (4)$$

rather than by explicitly integrating. The process of generating the $\mathcal{Y}^{(i)}$ is known as “Bayesian multiple imputation” when the sampled datasets are constructed using the “general location model” and its relatives [6].

3.1 The General Location Model

The general location model [6] is commonly used for imputing the values of missing data. This model divides the variables of interest $Y = [Y_1, Y_2, ..., Y_d]^{\text{T}}$ into a categorical portion $W = [W_1, W_2, ..., W_h]^{\text{T}}$ and a continuous portion $Z = [Z_1, Z_2, ..., Z_k]^{\text{T}}$. It is convenient to collapse the W_i into a single categorical variable W that takes on p different values, each representing a particular complete configuration of $[W_1, W_2, ..., W_h]^{\text{T}}$. For example, if we wish to model two variables $W_1 \in \{-1, 1\}$ and $W_2 \in \{\mathit{red}, \mathit{green}, \mathit{blue}\}$, we construct $W \in \{1, 2, ..., 6\}$ to represent all the elements of $\{-1, 1\} \times \{\mathit{red}, \mathit{green}, \mathit{blue}\}$. This encoding also facilitates modelling “structural zeros”, i.e. forbidden configurations: If *blue* cannot appear together with 1, we can shrink the domain of W to $\{1, 2, 3, 4, 5\}$. We will still use the individual W_i variables for conditioning: The domain of $W|(W_2 = \mathit{green})$ will have size two, for example, to represent the remaining possible configurations $\{(1, \mathit{green}), (-1, \mathit{green})\}$ that are consistent with $W_2 = \mathit{green}$. The continuous variables $Z = [Z_1, Z_2, ..., Z_k]^{\text{T}}$ may each take on any real value.

The general location model is a joint distribution over $Y = (W, Z) = [W, Z_1, Z_2, ..., Z_k]^{\text{T}}$ where:

$$\begin{aligned} W &\sim \text{Mult}(\boldsymbol{\theta}) \\ Z|(W = w) &\sim N(\boldsymbol{\mu}_w, \Sigma) \, . \end{aligned}$$

We model W using a multinomial distribution with a length-p parameter vector $\boldsymbol{\theta}$, $\boldsymbol{\theta} > 0, \sum_i \theta_i = 1$. We model Z given W as a multivariate normal random variable with the length-k mean vector $\boldsymbol{\mu}_w$ that is allowed to depend on W, and a shared $k \times k$ covariance matrix Σ. This is effectively an LDA model [3]; notice that the marginal distribution of Z is a mixture of p Gaussians where each component has mean $\boldsymbol{\mu}_w$ and covariance Σ. This describes the distribution of Y which represents a single training sample. The likelihood of a dataset $\mathcal{Y}$ of i.i.d. training samples is simply the product of the likelihoods of each of the samples.

3.1.1 Probability of Missing Data given Observed Data

We have described a probabilistic model $P(\mathcal{Y}|\Theta)$ of our data that allows us to compute the likelihood of a dataset $\mathcal{Y}$ given the parameters $\Theta = (\boldsymbol{\theta}, \boldsymbol{\mu}_w, \Sigma)$. In our problem, however, the data are partitioned into $\mathcal{Y} = (\mathcal{Y}_{\text{mis}}, \mathcal{Y}_{\text{obs}})$, the missing data and the observed data, and the quantity we really need is $P(\mathcal{Y}_{\text{mis}}|\mathcal{Y}_{\text{obs}})$. In principle, we can obtain this by putting a prior on Θ, conditioning on $\mathcal{Y}_{\text{obs}}$, and integrating out the parameters.

$$P(\mathcal{Y}_{\text{mis}}|\mathcal{Y}_{\text{obs}}) = \int P(\mathcal{Y}_{\text{mis}}|\mathcal{Y}_{\text{obs}}, \Theta) P(\Theta|\mathcal{Y}_{\text{obs}}) \mathrm{d}\Theta \qquad (5)$$

However, not only is (5) an intractable integral over Θ, it contains a term $P(\Theta|\mathcal{Y}_{\text{obs}})$ that is an intractable integral over $\mathcal{Y}_{\text{mis}}$. However we will show that if we knew $\mathcal{Y}_{\text{mis}}$ (and therefore $\mathcal{Y}$) then sampling from $\Theta|(\mathcal{Y}_{\text{obs}}, \mathcal{Y}_{\text{mis}})$ is straightforward, and that if we know Θ, then sampling from $\mathcal{Y}_{\text{mis}}|(\mathcal{Y}_{\text{obs}}, \Theta)$ is also straightforward. Therefore in order to overcome the intractability of (5), we use Markov Chain Monte Carlo sampling [3, 8] to obtain samples from the distribution $\mathcal{Y}_{\text{mis}}|\mathcal{Y}_{\text{obs}}$. In particular, we use a Gibbs-type Markov chain, where we alternately sample $\mathcal{Y}_{\text{mis}}|(\mathcal{Y}_{\text{obs}}, \Theta)$ using Equations (11,12), and $\Theta|(\mathcal{Y}_{\text{obs}}, \mathcal{Y}_{\text{mis}})$ using Equations (8,9,10):

$$\begin{aligned} \mathcal{Y}_{\text{mis}}^{(t+1)} &\sim \mathcal{Y}_{\text{mis}}|\Theta^{(t)}, \mathcal{Y}_{\text{obs}} \\ \Theta^{(t+1)} &\sim \Theta|\mathcal{Y}_{\text{mis}}^{(t+1)}, \mathcal{Y}_{\text{obs}} \; . \end{aligned}$$

Given a starting guess Θ^0 and enough iterations, the samples of $\mathcal{Y}_{\text{mis}}$ obtained will come from the true distribution of $\mathcal{Y}_{\text{mis}}|\mathcal{Y}_{\text{obs}}$. We then use these samples to create m complete data sets $\mathcal{Y}^{(1)}, \mathcal{Y}^{(2)}, ..., \mathcal{Y}^{(m)}$ which we use to approximate expectations over (5). In the following sections we sketch the derivation of the distributions necessary for Gibbs sampling in this context; for a more complete treatment, see Schafer [6].

3.1.2 Posterior over Parameters

To get $P(\Theta|\mathcal{Y}_{\text{obs}}, \mathcal{Y}_{\text{mis}})$ so that we can sample $\Theta^{(t+1)} \sim \Theta|(\mathcal{Y}_{\text{mis}}^{(t+1)}, \mathcal{Y}_{\text{obs}})$, we first place a non-informative, improper prior[2] over $\Theta = (\boldsymbol{\theta}, \boldsymbol{\mu}_w, \Sigma)$ as follows:

$$\begin{aligned} \boldsymbol{\theta} &\sim \text{Dir}(\boldsymbol{\alpha}) & (6) \\ P(\Sigma, \boldsymbol{\mu}_1, \boldsymbol{\mu}_2, ..., \boldsymbol{\mu}_p) &\propto |\Sigma|^{-\frac{k+1}{2}} \; . & (7) \end{aligned}$$

This gives a Dirichlet prior on $\boldsymbol{\theta}$, an improper uniform prior on $\boldsymbol{\mu}_w$, and the standard improper non-informative prior on Σ. (We typically take $\boldsymbol{\alpha} = \mathbf{1}$ to give a uniform prior on $\boldsymbol{\theta}$.) Using these priors, we can compute the posterior distribution of $\boldsymbol{\theta}$, $\boldsymbol{\mu}_j$, and Σ given n completely observed i.i.d. samples of Y, denoted $\mathcal{Y} = \{\boldsymbol{y}_1, \boldsymbol{y}_2, ..., \boldsymbol{y}_n\} = \{(w_1, \boldsymbol{z}_1), (w_2, \boldsymbol{z}_2), ..., (w_n, \boldsymbol{z}_n)\}$. We define $\boldsymbol{c}$ where $c_j = \#\{w_i \in \mathcal{Y} \text{ s.t. } w_i = j\}$ are the counts of data points that have W in configuration w_j, as well as the point estimates $\hat{\boldsymbol{\mu}}_j$ and $\hat{\Sigma}$:

$$\begin{aligned} \hat{\boldsymbol{\mu}}_j &= \frac{1}{c_j} \sum_{i:w_i=j} \boldsymbol{z}_i \\ \hat{\Sigma} &= \frac{1}{n} \sum_j \sum_{i:w_i=j} (\boldsymbol{z}_i - \hat{\boldsymbol{\mu}}_j)(\boldsymbol{z}_i - \hat{\boldsymbol{\mu}}_j)^{\mathrm{T}} \; . \end{aligned}$$

We now write the posterior distribution over the model parameters as:

$$\begin{aligned} \boldsymbol{\theta} &\sim \text{Dir}(\boldsymbol{\alpha} + \boldsymbol{c}) & (8) \\ \Sigma|\mathcal{Y} &\sim \text{Wish}^{-1}(n-p, \frac{1}{n}\hat{\Sigma}^{-1}) & (9) \\ \boldsymbol{\mu}_j|\Sigma, \mathcal{Y} &\sim N(\hat{\boldsymbol{\mu}}_j, \frac{1}{c_j}\Sigma) \; . & (10) \end{aligned}$$

The posterior $\boldsymbol{\theta}$ is again Dirichlet-distributed, the posterior Σ is inverse-Wishart distributed with $n-p$ degrees of freedom and scale matrix $\hat{\Sigma}^{-1}$, and the posterior $\boldsymbol{\mu}_j$ are multivariate normal. To sample from $\Theta|\mathcal{Y}_{\text{obs}}, \mathcal{Y}_{\text{mis}}$, we sample $\boldsymbol{\theta}$ according to (8), we sample a covariance $\tilde{\Sigma}$ from (9), and then we sample each $\boldsymbol{\mu}_j \sim N(\hat{\boldsymbol{\mu}}_j, \tilde{\Sigma})$ according to (10).

3.1.3 Posterior over Missing Data

To get $P(\mathcal{Y}_{\text{mis}}|\mathcal{Y}_{\text{obs}}, \Theta)$ so that we can sample $\mathcal{Y}_{\text{mis}}^{(t+1)} \sim \mathcal{Y}_{\text{mis}}|(\Theta^{(t)}, \mathcal{Y}_{\text{obs}})$, suppose $\boldsymbol{w}_{\text{obs}}, \boldsymbol{z}_{\text{obs}}$ are the observed categorical and continuous parts of a vector Y. Then

$$P(W = w|\boldsymbol{w}_{\text{obs}}, \boldsymbol{z}_{\text{obs}}, \Theta) \propto \begin{cases} \theta_w \cdot P(\boldsymbol{z}_{\text{obs}}|\boldsymbol{\mu}_w, \Sigma) \text{ where } w \text{ is consistent with } \boldsymbol{w}_{\text{obs}} \\ 0 \text{ otherwise .} \end{cases} \qquad (11)$$

[2] Informative priors are easily incorporated; see Schafer [6].

Note, for example, that if $\boldsymbol{w}_{\text{obs}}$ is completely observed then exactly one configuration w^* is consistent with $\boldsymbol{w}_{\text{obs}}$, and $P(W = w^*|\boldsymbol{w}_{\text{obs}}) = 1$. If $\boldsymbol{w}_{\text{obs}}$ is partly observed, then the probability of each consistent configuration is weighted by its "prior" probability θ_w, and by the likelihood that such a configuration would have generated $\boldsymbol{z}_{\text{obs}}$. The posterior distribution of Z_{mis}, the unobserved continuous variables, is given by

$$Z_{\text{mis}}|(W = w, \boldsymbol{z}_{\text{obs}}, \Theta) \sim N(\boldsymbol{\mu}_w(\boldsymbol{z}_{\text{obs}}), \Sigma(\boldsymbol{z}_{\text{obs}})) \ . \tag{12}$$

Here, $\boldsymbol{\mu}_w(\boldsymbol{z}_{\text{obs}})$ and $\Sigma(\boldsymbol{z}_{\text{obs}})$ are computed using standard formulas for conditioning in multivariate Gaussians. To obtain one sample of missing data, we first sample the missing discrete variables w from $W|(\boldsymbol{w}_{\text{obs}}, \boldsymbol{z}_{\text{obs}}, \Theta)$ according to Equation (11), and then we sample the missing continuous variables from $Z_{\text{mis}}|(W = w, \boldsymbol{z}_{\text{obs}}, \Theta)$.

4 Computing $\hat{Q}_{\text{obs}}$ and Assessing Confidence

We now have a procedure that takes a partially observed data set $\mathcal{Y}_{\text{obs}}$ and produces an estimate $\hat{Q}_{\text{obs}}$ of the optimal Q function. The procedure is as follows:

1. Generate m imputed datasets $\mathcal{Y}^{(1)}, \mathcal{Y}^{(2)}, ..., \mathcal{Y}^{(m)}$ from $\mathcal{Y}_{\text{obs}}$ using Gibbs sampling with the general location model
2. For each imputed dataset $\mathcal{Y}^{(i)}$, compute $\hat{Q}(o, a; \mathcal{Y}^{(i)})$ using batch Q-learning
3. Compute $\hat{Q}_{\text{obs}}(o, a; \mathcal{Y}_{\text{obs}}) \leftarrow \frac{1}{m} \sum_i \hat{Q}(o, a; \mathcal{Y}^{(i)})$

This technique will be unaffected by non-response bias when data are MAR and missingness is ignorable, and should perform better than complete case analysis (throwing out incomplete trajectories) in any setting where missing values are at least partially predictable from the observed data.

Given that we have chosen $\hat{Q}_{\text{obs}}$ as our estimate, we want a useful measure of our confidence in $\hat{Q}_{\text{obs}}$. We propose to use the bootstrap to approximate the training set distribution in order to assess the variability of $\hat{Q}_{\text{obs}}$, which we express in terms of how often an action a is estimated to be optimal given a particular observation o.

4.1 The Bootstrap

The bootstrap [2, 8] is a resampling procedure that can be used to obtain confidence measures by using the training set to approximate the true data distribution. To do this in the context of supervised learning, we construct b training sets of the same size as $\mathcal{Y}$ by uniformly randomly drawing training samples with replacement from $\mathcal{Y}$. Then for each bootstrapped training set, we compute our estimate of interest. These b estimates can then be used to determine the variance, percentiles, etc. of our estimator.

In our setting, we first construct bootstrapped datasets $\mathcal{Y}^{[1]}_{\text{obs}}, \mathcal{Y}^{[2]}_{\text{obs}}, ..., \mathcal{Y}^{[t]}_{\text{obs}}$ from $\mathcal{Y}_{\text{obs}}$. Then from each of these bootstrapped datasets we estimate $\hat{Q}_{\text{obs}}(o, a; \mathcal{Y}^{[i]}_{\text{obs}})$ using the imputation procedure given above, which gives us t Q-function estimates $\hat{Q}^{[1]}_{\text{obs}}, \hat{Q}^{[2]}_{\text{obs}}, ..., \hat{Q}^{[b]}_{\text{obs}}$. This means that for each $\mathcal{Y}^{[i]}_{\text{obs}}$, we impute m complete datasets $\mathcal{Y}^{[i](1)}, \mathcal{Y}^{[i](2)}, ..., \mathcal{Y}^{[i](m)}$, learn a $\hat{Q}(o, a; \mathcal{Y}^{[i](j)})$ for each one, and average these to obtain $\hat{Q}^{[i]}_{\text{obs}}$. Therefore a total of $b \times m$ applications of Q-learning are required to produce the b bootstrapped estimates.

The bootstrap is commonly used to assess the distribution of a scalar estimator, and in such cases it is common to report a standard error or a confidence interval or a similar one-dimensional quantity. When considering higher-dimensional estimators, however, choosing a useful measure of confidence requires some knowledge of how the estimator will be used. In the case of $\hat{Q}_{\text{obs}}$, we will typically reconstruct the greedy policy $\pi(o; \hat{Q}_{\text{obs}}) = \arg\max_{a'} \hat{Q}_{\text{obs}}(o, a')$ in hopes that it will be close in value to the optimal policy π^*. We therefore choose a measure of confidence that conveys how the variability of $\hat{Q}_{\text{obs}}$ induces variability in $\pi(o; \hat{Q}_{\text{obs}})$. We do this by letting each $\hat{Q}^{[i]}_{\text{obs}}$ "vote" for the best action at each setting of o; each of the t votes is the greedy action given by $\pi(o; \hat{Q}^{[i]}_{\text{obs}})$. By examining the proportion of the $\hat{Q}^{[i]}_{\text{obs}}$ that voted for each action, one can assess how well the training data support each action choice. In the next section we present a case study that illustrates this procedure in more detail. This approach is discussed by Efron and Tibshirani [1], who show that the proportion of votes for each action estimates the posterior probability that we would decide that a was the optimal action under a uniform prior[3] on $\hat{Q}(o, a)$.

5 Case Study: STAR*D

The Sequenced Treatment Alternatives to Relieve Depression (STAR*D) study [5] is a randomized clinical trial in which patients received a sequence of treatments for depression. It was designed to first identify patients who do not respond to a first-line antidepressant (termed "treatment-resistant"), and then evaluate different sequences of treatments for this subgroup. After this first "level" of treatment, up to three more treatments were offered to the treatment-resistant subgroup.

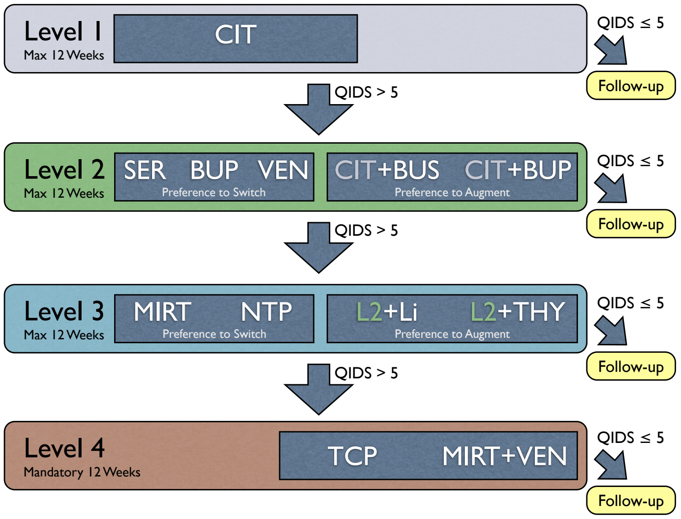

Figure 1: Schematic diagram of STAR*D. Dark grey boxes are sets of possible treatments. QIDS measures symptom relief.

At levels 2 and 3, the patient was allowed to choose a subset of "switch" treatments where a new drug was substituted for the previous one, or a set of "augment" treatments where the new drug was added to the current treatment. Next, the patient was randomized to one of the treatments in the selected preference class. Level 4 had one set of treatments.

As the patients progressed through the study, measurements were made of their symptom relief and side effects at weeks 0, 2, 4, 6, 9, and 12 within each level. Symptom relief, measured by the Quick Inventory of Depressive Symptoms (QIDS, range 0-27, lower is better) was used to determine if a patient had remitted (defined as QIDS ≤ 5) or not. If a patient remitted in a level, they proceeded to a follow-up phase where no further subsequent treatments were offered. If a patient did not remit, they were to continue on to the next level. While patients were encouraged to remain in each level for the full 12 weeks, they were permitted to move to the next level early if they felt their results were unsatisfactory. Figure 2 illustrates how patients progressed through the study[4]. From a reinforcement learning perspective, the STAR*D data represent a collection of sample trajectories of agents moving through observation space, following a random policy at each step. The observation space is potentially very rich; if we include all within-level measurements, we could have a 24-dimensional vector of observations describing the patients' symptoms and side-effects. On the other hand, the maximum trajectory length of a treatment-resistant patient is 3. Furthermore, the possible actions are different at each level, and at levels 2 and 3 they depend on patient preference.

Although this setting is somewhat atypical in the RL field, we can apply the same batch methods as we would in other more familiar settings. The only thing not explicitly defined by the STAR*D data is reward. Developing a meaningful reward for STAR*D is a complex question in itself, but for this study we consider only symptom relief as measured by QIDS. In the study, having a QIDS score ≤ 5 was defined as a successful outcome, i.e. remission, and we want our reward function to reflect this. Define $\text{QIDS}_{\text{last}}$ to be the last QIDS score measured within the level in question, i.e. the last one before the patient either goes to follow up, or continues on to the next level, or (at level 4) finishes the study.

[3] They also note that a uniform prior may not be the best choice, given that the probability estimates produced by the bootstrap do not necessarily give confidence statements with good frequentist properties.

[4] This is a simplified view; further details on the study and treatments are described by Rush et al. [5].

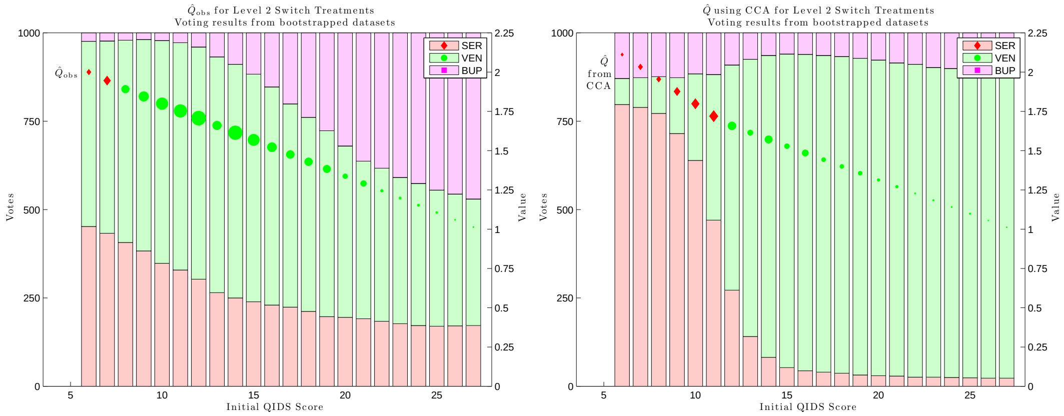

Figure 2: Q-functions learned from STAR*D data using multiple imputations (left) and CCA (right.) Value functions learned from the original $\mathcal{Y}_{\mathrm{obs}}$ and CCA datasets are indicated by the markers; votes for each action from bootstrapped datasets are shown as bars. Marker sizes indicate the number of training trajectories with each Initial QIDS.

We define the reward obtained in a level by administering each treatment to a patient to be a function only of their $\mathrm{QIDS}_{\mathrm{last}}$: The reward is the *proportion* of non-remitted patients in a reference distribution who have a QIDS that is *higher* (worse) than $\mathrm{QIDS}_{\mathrm{last}}$. Therefore, the lower a patient's $\mathrm{QIDS}_{\mathrm{last}}$, the higher the reward. We use an empirical distribution of baseline QIDS based on the data, which makes the reward linear in the *percentile* of $\mathrm{QIDS}_{\mathrm{last}}$. The per-level reward is bounded above by 1.0, which is obtained when the patient remits, because at remission the patient has a QIDS score better than all of the patients in the reference distribution since each of them has QIDS > 5. Finally, if a patient remits in level ℓ, and therefore will not enter further levels which would generate more reward, an additional reward of $4 - \ell$ is added so that early remission is not penalized.

5.1 Missing Data in STAR*D

Given this definition, we could in theory assign a reward to each action (treatment) of each trajectory (patient record) in the STAR*D data, and then use batch Q-learning to estimate an optimal Q-function and policy. However, the STAR*D data have a significant amount of missingness, because many patients did not follow the protocol as defined above. In reality, many patients dropped out of the study prior to remission. For these patients, we have observations from when they enter the study up until the time they drop out, and none thereafter. There are many reasons for drop-out that one might imagine: Perhaps the side-effects were too severe, or the patient felt that they were cured despite having a QIDS > 5, or they felt the clinic visits were too demanding, or they became pregnant, or any number of other reasons. Some of these, like side-effect burden, we can observe in the data; others, like feeling the clinic visits are too demanding, we cannot.

By comparing the observations we do have from patients who dropped out with those who did not drop out, we have found that the STAR*D data are *not* MCAR, and we are therefore very reluctant to trust the results of CCA. On the other hand, we have observed data that are predictive of drop-out, like the side-effects measurements. We therefore use multiple imputations from the general location model to build a Q-function that is conditioned on the entire observed portion of the STAR*D data in hopes that we can explain some of the drop-out and obtain a less biased estimate of the true Q-function.

5.2 Our Analysis

To date, our analysis of the STAR*D data has focused on levels 2 and 3. We use Q-learning to build optimal value functions, with the first QIDS measured in a level as our input observation o, and the reward function based on $\mathrm{QIDS}_{\mathrm{last}}$ as described above. A separate linear model is built for each $(\mathrm{preference}, \mathrm{treatment})$ pair; that is, for each preference ("switch" or "augment") and treatment, we estimate a linear function that relates the initial QIDS score of a patient to that patient's expected total future reward. We can then maximize over treatments to obtain a piecewise linear value function dependent on initial QIDS, as well as the greedy policy based on that value function.

As an example, Figure 2 shows the Q-functions learned at level 2 for patients who preferred to switch treatments. On the left is the result of our $\hat{Q}_{\text{obs}}$ approach using multiple imputations, and on the right is the result of Q-learning on a CCA dataset, where we have thrown out all of the patients for whom we had incomplete data. The voting results of 1000 bootstrapped datasets are shown in each case by the bars behind the value function. Note that although we are only using QIDS in these models, our imputation model includes nearly everything that is recorded about a patient in an effort to explain drop-out as well as we can given the available data. This a major advantage of our approach; even though (for now at least) we will only use a few variables in our Q-learning analyses, we can use many more variables in the imputation model to help ensure that the resulting imputed datasets effectively represent the missing data we care about and help avoid non-response bias.

One can see in the figures how confident we are in each of our action choices. Our model prefers either SER or VEN for patients with lower QIDS, and any of the three treatment options for patients with higher QIDS. However, in each of these cases there is no clear winner as none of the treatments has a large majority of the votes; however, for low QIDS, BUP is a clear loser. The CCA analysis on the other hand is much more confident that SER is the right choice for low initial QIDS, and VEN is the right choice for high QIDS. This level of confidence is not valid in general because, as we mentioned, the characteristics of the patients that remain in the CCA dataset are very different from those of the entire dataset. For example, further analysis has shown that for patients with high initial QIDS, the level 2 reward for VEN is better than that for SER and BUP in the CCA data. However, among patients who later dropped out and are therefore not present in the CCA data, VEN is *worse* than SER and BUP. The non-response bias illustrated by this inversion at least partially explains the difference between using multiple imputations and using CCA for these data.

6 Conclusion

We have shown how methods for missing data can be applied to batch reinforcement learning. We have explained why reasoning about missing data is important in theory and shown its effect in practice on the STAR*D dataset. We will continue this line of research in several directions: We are interested in alternatives to the estimator $\hat{Q}_{\text{obs}}$; for example we may wish to use a posterior mode rather than the posterior mean in some situations. We are also interested in a combined Bayesian approach that would not separate the missing data inference from the Q-learning, as such a formulation would give full posterior distributions over action choices. Finally, we are interested in more flexible, semi-parametric imputation models for cases where we have more training samples.

Acknowledgments

We acknowledge support for this project from National Institute of Health grants RO1 MH080015 and P50 DA10075. We also thank the STAR*D research team (`http://www.star-d.org/`).

References

[1] Bradley Efron and Robert Tibshirani. The problem of regions. *The Annals of Statistics*, 26(5):1687–1718, 1998.

[2] Bradley Efron and Robert J. Tibshirani. *An Introduction to the Bootstrap*. Chapman & Hall/CRC, 1993.

[3] Trevor Hastie, Robert Tibshirani, and Jerome Friedman. *The Elements of Statistical Learning*. Springer, 2002.

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

[5] A. J. Rush and M. Fava et al. Sequenced treatment alternatives to relieve depression (STAR*D): rationale and design. *Controlled Clinical Trials*, 25(1):119–42, Feb 2004.

[6] Joseph L. Schafer. *Analysis of Incomplete Multivariate Data*. CRC Press, 1997.

[7] R. S. Sutton and A. G. Barto. *Reinforcement Learning: An Introduction*. MIT Press, 1998.

[8] Larry Wasserman. *All of Statistics: A Concise Course in Statistical Inference*. Springer, 2004.