Assessing Confidence in Policies Learned from Sequential Randomized Trials

Daniel J. Lizotte, Eric Laber, and Susan A. Murphy

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

Abstract. Sequential randomized trials are becoming an increasingly important mechanism for gathering data to aid in clinical decision making, particularly in the study of chronic illnesses. The sequential nature of the resulting data necessitates an analysis that respects the principles of optimal sequential decision making. Reinforcement learning provides several well-studied algorithms that are appropriate for this type of data, but most of these do not provide measures of confidence in the learned policy. We present a regularized voting procedure based on the bootstrap that enables us to assess our confidence in the stability of our choice of optimal treatment, i.e., our confidence that we would discover the same optimal treatment if we were to re-run the study and the analysis. These confidence measures help us to identify when a subset of treatments are essentially equivalent. We present an example of this type of analysis using data from the Sequenced Treatment Alternatives to Relieve Depression (STAR*D) clinical trial.

1 Introduction

In a sequential randomized trial [1], participants are offered a sequence of treatments over the course of the trial which consists of a series of k *stages*. As in a traditional one-stage randomized trial, each offering is selected uniformly randomly from a small set of appropriate treatments. As each participant progresses through the study, observations are recorded that will later allow us to assess which treatment or treatments are most preferable for different types of patients at each stage. Experimenters record both observations related to treatment outcomes (e.g. symptom relief and side-effects) and also observations which we may later use to tailor treatments to specific patients (e.g. demographic information and concomitant disorders.) Of course, some observations may serve both purposes; for example, the treatment offered at the next stage could be chosen based on the efficacy of the treatment offered at the previous stage.

At the end of such a trial, we have a dataset $\mathcal{T}$ consisting of *trajectories* $t_1, t_2, ...t_n$, one for each of n participants. These are analogous to n *training examples* in the supervised machine learning setting; we use the term "trajectory" here to emphasize the sequential nature of the acquired data. We call $\mathcal{T}$ our *training set.* Given $\mathcal{T}$, we would like to recover an optimal *policy* that completely specifies which treatment we should offer a new patient, conditioned on our most current information about that new patient. This information will be of the same type as the observations made during the sequential randomized trial. A policy is "optimal" relative to an outcome (e.g. time to remission) if it optimizes the expected future value of that outcome.

Recovering an optimal policy from data is a well-studied problem in artificial intelligence, addressed in particular by the field of reinforcement learning. However, commonly used methods, including *fitted Q iteration* [2] which we use in our example, do not provide any measure of solution stability, i.e. they do not indicate whether we would discover the same optimal treatment if we were to re-run the study and the analysis. This gives cause for concern, particularly when $\mathcal{T}$ is small relative to the number of features used in learning and overfitting is a danger. One technique used to assess supervised learning techniques that is applicable in our situation is the bootstrap [3, 4]; however, we will show that naïve application of the bootstrap can result in inconsistent estimators of treatment choice stability when treatments are equivalent. We then give a novel modification of the bootstrap called the "adaptive bootstrap" – a voting mechanism that avoids the inconsistency problem in this case. We then illustrate its use on data from the Sequenced Treatment Alternatives to Relive Depression (STAR*D) study.

1.1 Reinforcement Learning

We now formalize the idea of optimizing expected outcomes using the language of Reinforcement Learning (RL) [5]. The problem of recovering optimal policies from data has been studied extensively by the RL community. In the field of reinforcement learning, treatments are termed *actions*, outcome and tailoring observations are together termed *observations*, and *rewards* are a function of the next observation in the sequence. The n training trajectories $t_1, t_2, ..., t_n$ in our dataset $\mathcal{T}$, from a reinforcement learning perspective, are each a sequence of the form $(o_1, a_1, r_1, o_2, a_2, r_2, ..., o_k, a_k, r_k, o_{k+1})$ comprised of the observations, actions, and rewards, starting from stage 1 and continuing to stage k.

Given this data, reinforcement learning seeks to recover an optimal *policy*. A policy π is a function[1] that maps the space of observations to the space of actions. (I.e., it maps the space of patient observations to the space of treatments.) The *value* of seeing an observation o_i, taking an action a and then following a policy π (i.e. choosing $a_j = \pi(o_j)$ for $i+1 \leq j \leq k$) is defined to be the expected sum of rewards $r_i, r_{i+1}, ..., r_k$ obtained using this strategy. Each reward r_j is a specified function of the tuple (o_j, a_j, o_{j+1}). The expected sum of rewards is denoted $Q_i^\pi(o_i, a)$, which is given by

$$Q_i^\pi(o_i, a) = \mathbb{E}_{O_{i+1}|o_i,a}[r_i + Q_{i+1}^\pi(O_{i+1}, a)] \tag{1}$$

Here, the expectation is taken over $O_{i+1}|o_i, a_i$ which is the conditional distribution of O_{i+1} given the realizations $A_i = a_i, O_i = o_i$, as defined by the underlying dynamics of the system. In the clinical trial setting, these dynamics define how a participant responds to different treatments. A policy π^* is optimal if and only if [5]

$$\max_a Q_i^{\pi^*}(o, a) \geq \max_a Q_i^\pi(o, a) \text{ for all } i, o \text{ and } \pi. \tag{2}$$

[1] More generally, π could be a collection of conditional distributions to allow random action choices at each step. For example, during a sequential randomized trial, we effectively follow a policy $\pi_{\text{rand}}|o$ that is uniformly distributed over possible actions.

To see how we might estimate an optimal policy, consider the final stage, where we take one last action. At stage k, we have

$$Q_k(o_k, a) = \mathbb{E}_{O_{k+1}|o_k,a}[r_k]. \tag{3}$$

This quantity does not depend on π, since a is the last action in the trajectory; we therefore write Q_k with no superscript. Suppose we can estimate $Q_k(o_k, a)$ well for any o_k and a. We could accomplish this by regressing r_k on o_k and a using our dataset of trajectories $\mathcal{T}$, giving an estimated Q_k function $\hat{Q}_k(o_k, a; \mathcal{T})$. We then estimate the optimal stage-k action as follows:

$$\hat{\pi}^*(o_k) \triangleq \operatorname*{argmax}_a \hat{Q}_k(o_k, a; \mathcal{T}). \tag{4}$$

We choose the action that gives us the highest expected reward according to our estimate $\hat{Q}_k$. Having chosen the final action $a_k = \hat{\pi}^*(o_k)$, we can now estimate $Q^{\hat{\pi}^*}_{k-1}(o_{k-1}, a; \mathcal{T})$ by regressing $r_{k-1}+\hat{Q}_k(o_k, \hat{\pi}^*(o_k); \mathcal{T})$ on o_{k-1} and a, again using the trajectories from $\mathcal{T}$, giving $\hat{Q}^{\hat{\pi}^*}_{k-1}(o_{k-1}, a; \mathcal{T})$. We then estimate the optimal action at stage $k-1$ by

$$\hat{\pi}^*(o_{k-1}) \triangleq \operatorname*{argmax}_a \hat{Q}^{\hat{\pi}^*}_{k-1}(o_{k-1}, a; \mathcal{T}). \tag{5}$$

We continue backward through the stages in this manner, alternately computing $\hat{Q}^{\hat{\pi}^*}_i(o_i, a; \mathcal{T})$ by regression and then $\hat{\pi}^*(o_i)$ by maximization for $k \geq i \geq 1$. This process in general is known as *fitted Q iteration* [2]. Various regression models could be be used to construct the $\hat{Q}_i$ depending on the particular application we have in mind, e.g. lookup tables, linear regression, neural networks, etc. For the remainder of the text we will drop the superscript $\hat{\pi}^*$ and write $\hat{Q}_i(o_i, a; \mathcal{T})$ to represent an estimated optimal Q-function constructed in this manner from a dataset $\mathcal{T}$, since we can recover $\hat{\pi}^*(o_i)$ as needed using the argmax operator.

1.2 The Bootstrap

Originally introduced by Efron [6], the bootstrap is a resampling method that simulates samples from the true data generating distribution F by sampling instead from the empirical distribution $\hat{F}_n$ defined by our observed dataset $\mathcal{T}$. We assume the training trajectories in $\mathcal{T}$ are independent and identically distributed (IID), drawn from the distribution F, so that $T_1, T_2, ..., T_n \sim F$. We define the empirical distribution $\hat{F}_n$ to have point mass $1/n$ at each of the the n trajectories in $\mathcal{T}$. Having done so, we define a *bootstrap sample* $\mathcal{T}^*$ to be a sample of n trajectories from $\hat{F}_n$, i.e. $T_1^*, T_2^*, ..., T_n^* \sim \hat{F}_n$. Such a dataset is constructed by sampling from the trajectories in $\mathcal{T}$ uniformly randomly with replacement.

The main utility of the bootstrap lies in its use in approximating quantities that would otherwise require samples from (or explicit knowledge of) F, the data generating distribution. We now show how it can be used to estimate the probability that our fitted Q iteration procedure will decide that a particular action is optimal.

2 “Voting Estimators”: Bootstrap Estimates of Indicator Averages

Suppose we have a dataset $\mathcal{T}$ from a 1-stage trial (i.e. $k = 1$) that consists of n pairs $t_i = (a_1^i, r_1^i)$. Here, $a_1^i \in \{1, 2\}$ and $r_1^i \in \mathbb{R}$. There are no observations in this example and it has only one stage, so we have one Q-function that depends only on a_1 which we write as $Q_1(a_1)$. We simply wish to determine which action produces the higher expected reward. Fitted Q iteration would tell us to first compute $\hat{Q}_1(1; \mathcal{T})$ and $\hat{Q}_1(2; \mathcal{T})$ using sample averages for example, and then choose the argmax over $a \in \{1, 2\}$ as the optimal action. Suppose then that we wish to assess the stability of our action preference: What is the probability that, given a new dataset $\mathcal{T} \sim F$, we would find action 1 to be optimal? Call this quantity $\nu(1)$. Estimating $\nu(1)$ is an instance of the “problem of regions” [7]. If we were able to sample b datasets $\mathcal{T}_1, \mathcal{T}_2, ..., \mathcal{T}_b$ from F, we could estimate $\nu(1)$ by learning a $\hat{Q}_1$ for each one, choosing the optimal action as described above, and finding the proportion of the b datasets that “voted” for action 1. Our estimate of the probability of preferring action 1 is therefore

$$\hat{\nu}_b(1) = \frac{1}{b} \sum \mathbf{1}[\hat{Q}_1(1; \mathcal{T}_i) - \hat{Q}_1(2; \mathcal{T}_i) > 0] \tag{6}$$

where $\mathbf{1}(p) = 1$ if p is true, and 0 otherwise. Of course we cannot sample b datasets from F; this would entail running b more sequential randomized trials. We could, however, draw b bootstrapped datasets $\mathcal{T}_1^*, \mathcal{T}_2^*, ..., \mathcal{T}_b^*$ from $\hat{F}_n$ and use

$$\hat{\nu}_b^*(1) = \frac{1}{b} \sum \mathbf{1}[\hat{Q}_1(1; \mathcal{T}_i^*) - \hat{Q}_1(2; \mathcal{T}_i^*) > 0] \tag{7}$$

as our estimate. One uses this procedure in the hope that $\hat{F}_n \approx F$, and that therefore $\hat{\nu}_b^*(1) \approx \hat{\nu}_b(1)$, which itself converges to $\nu(1)$ as b grows. Unfortunately, however, this hope is unfounded when $Q_1(1) - Q_1(2)$, the difference in the true Q_1 values, is at or near zero. We illustrate this problem using a simple example.

Suppose that r_1 is normally distributed with variance 1 and mean 0, independent of a_1. In this scenario, the action choice has no impact on the expected value of r_1, and $Q_1(1) - Q_1(2) = 0$. Given this generative model, we can compute $\hat{\nu}_b(1)$ for various b and see that it converges to 0.5 as one would expect given the law of large numbers. However, if we turn our attention to $\hat{\nu}_b^*(1)$ the situation is much worse. Suppose we examine the distribution of $\hat{\nu}_b^*(1)$ for $b = 1000$ as follows: We draw 100 datasets $\mathcal{T}_1, \mathcal{T}_2, ...\mathcal{T}_{100}$ from F, each with 10000 trajectories, 5000 using action 1 and 5000 using action 2. For each $\mathcal{T}_i$, we draw $b = 1000$ bootstrapped datasets $\mathcal{T}_{i,1}^*, \mathcal{T}_{i,2}^*, ...\mathcal{T}_{i,1000}^*$, each with 10000 trajectories and use them to compute $\hat{\nu}_b^*(1)$. Figure 1 (top) shows the resulting distribution of $\hat{\nu}_b^*(1)$: it is essentially uniform across the 100 datasets. Even though the true probability of finding action 1 to be optimal is $\nu(1) = 0.5$, our bootstrap estimate $\hat{\nu}_b^*(1)$ is equally likely to be anywhere between 0 and 1, and provides us with no information about the value of $\nu(1)$. Worse, $\hat{\nu}_b^*(1)$ is likely to lead us to believe that one action is preferable to the other even when no such evidence exists.

This problem, a symptom of “non-regularity”, arises in various situations where averages of indicators or other non-smooth functions of the data are estimated using the bootstrap [3, 8]. Furthermore, it is not alleviated by increasing n or b; the bootstrap does not produce a consistent estimator in these settings when there is significant mass concentrated at the point of non-smoothness—in our case, when $Q_1(1) - Q_1(2) = 0$. This problem can be mitigated, however, by modifying the indicator function to "regularize" it as we now show.

2.1 Regularized Voting

For convenience, we define $\Delta(1,2) \triangleq (Q_1(1) - Q_1(2))$, $\hat{\Delta}(1,2) \triangleq (\hat{Q}_1(1;\mathcal{T}) - \hat{Q}_1(2;\mathcal{T}))$, and $\hat{\Delta}_i^*(1,2) \triangleq (\hat{Q}_1(1;\mathcal{T}_i^*) - \hat{Q}_1(2;\mathcal{T}_i^*))$. Using this notation, our original bootstrap estimator is given by $\hat{\nu}_b^*(1) = \frac{1}{b}\sum \mathbf{1}[\hat{\Delta}_i^*(1,2) > 0]$. We now introduce a regularized estimator $\check{\nu}_b^*(1)$ that behaves properly when $\Delta(1,2) \approx 0$:

$$\check{\nu}_b^*(1) = \frac{1}{b}\sum \mathbf{1}\left[\hat{\Delta}_i^*(1,2) - \hat{\Delta}(1,2) + \hat{\Delta}(1,2)\cdot \mathbf{1}\left[\frac{|\hat{\Delta}(1,2)|}{\hat{\mathrm{se}}(\hat{\Delta}(1,2))} > z_{\alpha/2}\right] > 0\right] \tag{8}$$

The outermost indicator function in this estimator imposes different requirement on $\hat{\Delta}_i^*(1,2)$ to obtain a "1" vote from a bootstrap sample. Rather than requiring it be larger than zero, as $\hat{\nu}_b^*(1)$ does, this estimator can require instead that $\hat{\Delta}_i^*(1,2)$ be larger than $\hat{\Delta}(1,2)$, the difference in Q_1 values estimated from the original dataset $\mathcal{T}$, effectively re-centering the indicator function at $\hat{\Delta}(1,2)$. This re-centering is adaptive: It is imposed when the inner indicator function is "off", that is, when $\hat{\Delta}(1,2)/\hat{\mathrm{se}}(\hat{\Delta}(1,2))$ is small. Here, $\hat{\mathrm{se}}$ is an estimate of the standard error which is also computed using the bootstrap [9]. If $\hat{\Delta}(1,2)$ is approximately normal, then this ratio is approximately normal with unit variance and will be small when there is not sufficient evidence to support the belief that $\Delta(1,2) \neq 0$. On the other hand, if there is sufficient evidence to believe $\Delta(1,2) \neq 0$, we do not need to regularize, the third term will cancel with the second term, and $\check{\nu}_b^*(1) = \hat{\nu}_b^*(1)$. Note that we can easily extend (8) from a max operator over two alternatives to a max operator over m alternatives by expressing the max as m products of $(m-1)$ regularized indicators, of which at most one will be on at a time[2].

Figure 1 (bottom) shows the distribution of $\check{\nu}_b^*(1)$ using our simple example. Its distribution is concentrated around $\nu(1) = 0.5$, as it should be. We chose $z_{\alpha/2} = 1.96$, meaning we would expect about 0.05 of our estimators to be non-regularized, and therefore equivalent to $\hat{\nu}_b^*(1)$. In our test experiment, 7 of our 100 datasets had $|\hat{\Delta}(1,2)|/\hat{\mathrm{se}}(\hat{\Delta}(1,2)) > z_{\alpha/2}$, each of which resulted in a $\check{\nu}_b^*(1)$ far from 0.5.

This example tells us that any estimator of the same form as (8) will provide the regularization we need so long as the third term in (8) goes to zero when $\hat{\Delta}(1,2)$ is close to zero. Using this as our guideline, we have chosen one such estimator – the "adaptive bootstrap" estimator – that appears to work well in practice.

[2] In situations where there is not sufficient evidence for any of the alternatives to be the maximum, none of the indicators in the regularized max operator will be on. In such cases we count a $1/m$ of a vote for each alternative.

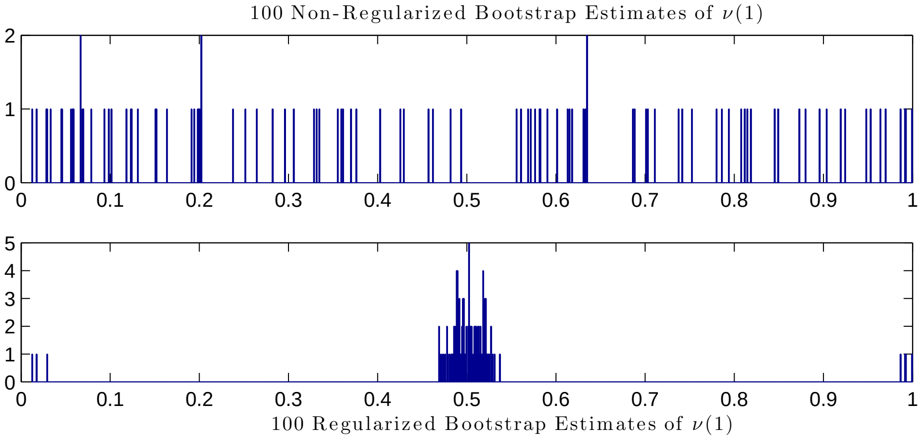

Fig. 1. Comparison of histograms of $\hat{\nu}_b^*(1)$ (top) and $\tilde{\nu}_b^*(1)$ (bottom).

3 Case Study: STAR*D

The Sequenced Treatment Alternatives to Relieve Depression (STAR*D) study [10] is a four-stage sequential randomized clinical trial in which participants received different treatments for depression. (In this study the word "level" is used to describe a stage.) In the Level 1, all participants were offered the same first-line antidepressant. Those who remitted left the study, and the remainder stayed on for up to three more Levels of treatment, with progression to the next Level always contingent on non-remission. At Levels 2 and 3, each participant 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. Participants were only randomized to treatments compatible with their selected preference class. In Level 4 there were no preference classes and participants were randomized to one of two treatments.

As the participants 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 participant had remitted (defined as QIDS ≤ 5) or not. If a participant remitted in a level, they proceeded to a follow-up phase where no further treatments were offered. If a participant did not remit, they were to continue on to the next level. While participants 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[3].

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. However, the STAR*D data have a significant amount of missingness, because many participants did not follow the protocol originally set forth. In

[3] This is a highly simplified description; further details on the study and treatments are described by Rush et al. [10].

reality, many participants dropped out of the study prior to remission. For these participants, we have observations from when they enter the study up until the time they drop out, and none thereafter. By comparing the observations we do have from participants who dropped out with those who did not drop out, we have found that the STAR*D data are *not* missing completely at random (MCAR) [11]. On the other hand, we have observed data that are predictive of drop-out, like the side-effects measurements. We therefore use a multiple imputation procedure [11] to build a Q-function that is conditioned on the entire observed portion of the STAR*D data, thus avoiding bias induced by drop-out.

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 example we consider only time to remission (in weeks) as determined by QIDS score. Of course, time to remission is a quantity we would like to minimize, not maximize; we will therefore construct our rewards to reflect *negative* time to remission and maximize their expectation. We are interested in those participants who remit within 30 weeks of starting their Level 2 treatment, and define $\mathrm{TTR}_{30} \triangleq \max(\text{Time to remission after L2}, 30)$; we will use the negative of this quantity to construct our rewards.

Table 1. Observations, actions, and rewards used in the STAR*D analysis.

	Level 2	**Level 3**
Observation	Last QIDS observed in Level 1 Switch/Augment preference	Last QIDS observed in Level 2 Switch/Augment preference
Action	Switch: SER,VEN,BUP Augment: +BUP,+BUS	Switch: MIRT, NTP Augment: +Li,+THY
Reward	Remitted in L2: $-\mathrm{TTR}_{30}$ Otherwise: 0	Remitted in or after L3: $-\mathrm{TTR}_{30}$ Otherwise: 0

Table 1 describes the observations, actions, and rewards we use at each of the two stages we consider in this analysis. We limit our use of observations to the Switch/Augment preference and the most recent QIDS score of a participant in order to explore how the most recent QIDS score can be used to tailor treatments to patients. Using these definitions, we build $\hat{Q}$ functions as described in Sect. 1.1 using linear regression, and we also construct estimates of the probabilities $\nu(a)$ for each action using both the regularized adaptive bootstrap and the non-regularized bootstrap estimators presented in Sect. 2.

Figure 2 shows the results of our analysis for participants at Level 2. The top row of graphs shows the analysis for patients with a preference to Augment treatment, and the bottom row shows the analysis for those with a preference to Switch treatment. Each graph shows the estimated optimal cost (i.e. negative value) marked in black diamonds as a function of the observed QIDS score at the end of Level 1. The size of each diamond indicates how many observations we have at different points on the QIDS axis. Behind the diamonds are stacked bar graphs indicating what proportion of the $b = 1000$ bootstrapped datasets voted for each action. The left column uses the $\hat{\nu}_b^*$ estimator, while the

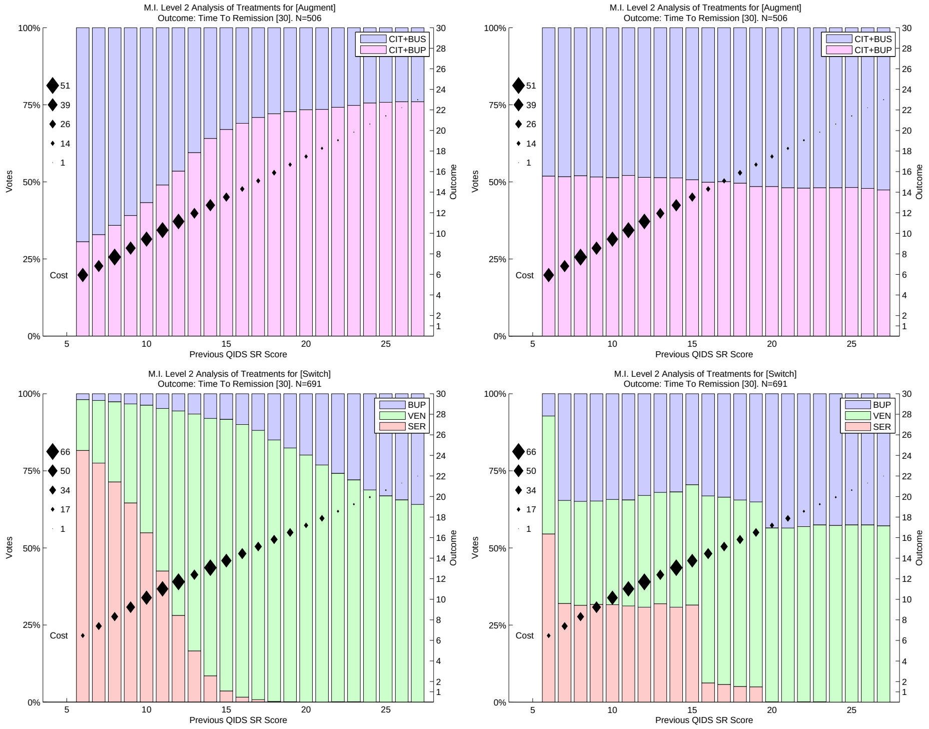

Fig. 2. STAR*D Level 2 Augment analyses with TTR_{30} rewards, presented here as costs. Votes on the left are non-regularized; votes on the right are regularized with the adaptive bootstrap.

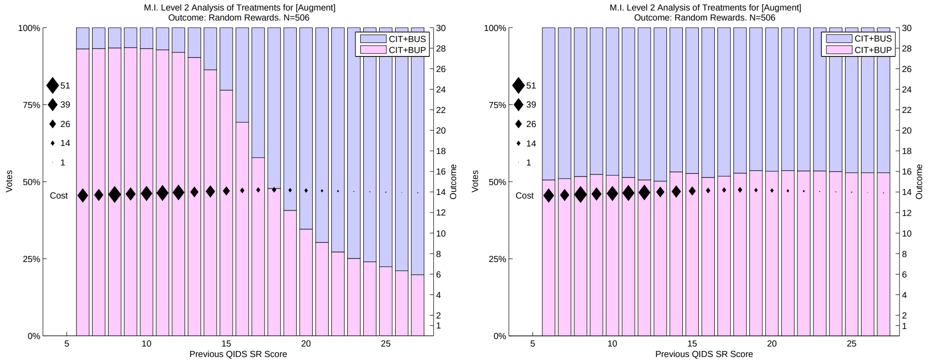

Fig. 3. Example analyses with **random** rewards drawn from a normal distribution with $\mu = 15$ and $\sigma = 5$. Votes on the left are non-regularized; votes on the right are regularized with the adaptive bootstrap.

right column uses the $\check{\nu}_b^*$ estimator. One can see that for the Augment group, although there appeared initially to be stable preference for the CIT+BUS action for participants with lower QIDS, this predicted stability does not persist if we use the adaptive bootstrap estimator, suggesting that the evidence for the preference is not very strong. For the Switch group, we again see significant effects from regularization, though a stable preference against BUP for very low QIDS and against SER for very high QIDS persists even when using the adaptive bootstrap. Based on this analysis, our Level 2 policy for those with a Switch preference would be:

$$\pi^*(\text{Switch}, \text{QIDS}) = \begin{cases} \text{SER or VEN} & \text{if QIDS} \leq 6 \\ \text{SER or VEN or BUP} & \text{if } 7 \leq \text{QIDS} \leq 15 \\ \text{VEN or BUP} & \text{if } 16 \leq \text{QIDS} \leq 27 \end{cases} \tag{9}$$

Figure 3 shows part of a simulated analysis that uses the same observations from the STAR*D dataset but with randomly generated rewards. All rewards for all actions at all stages for the example shown were normally distributed with mean 15 and standard deviation 5. Thus, we would expect all actions to appear to be equivalent and we would hope that any estimate of treatment preference stability would reflect this. The non-regularized voting estimator, shown on the left, shows a very strong (and incorrect) stability of the preference for CIT+BUP for low QIDS and a strong stability of the preference for CIT+BUS for high QIDS. These are even stronger than the non-regularized preference stabilities shown in the real data. The adaptive bootstrap estimator shown on the right correctly estimates the probability of preferring one action over the other to be approximately 0.5.

4 Discussion and Future Work

We have presented a method for assessing the stability of the optimal action choice made by applying fitted Q iteration to a dataset $\mathcal{T}$. We have shown how the naïve approach to estimating stability can lead to erroneous results when actions have equal expected reward, and presented the adaptive bootstrap estimator as a regularized alternative to the naïve estimator that produces correct results in this case.

Our motivation for this work stems from the fact that we are using training sets that are very expensive to collect, and thus are small relative to the signal (and effect) size. We therefore need robust measures of confidence to ensure that our findings are not spurious. Furthermore, it is a priori very likely for there to be no real difference in treatment effect in a sequential randomized trial, simply because the goal of a sequential randomized trial is to differentiate between treatments that have already been shown to be clinically useful, i.e. that we are in a state of clinical equipoise. Thus, when reasoning about data from sequential randomized trials, we want strong protection from problems that arise when the difference in expected treatment effects is zero.

Achieving this protection comes at a cost, however. Our method is aggressive in its regularization. While we gain in that we now correctly estimate $\nu(1)$ when $\Delta(1,2) = 0$, we also lose in that we will not be able to detect cases where $\Delta(1,2)$ is non-zero but small relative to the standard error of our estimate. However, as we mentioned, any estimator that causes the third term in (8) to go to zero when $\hat{\Delta}(1,2)$ goes to zero will provide protection against non-regularity as n grows. We may be able to construct a more "gentle" regularized estimator by replacing the inner indicator function of (8) with a continuous function that smoothly interpolates between regularizing at $\hat{\Delta}(1,2) = 0$ and non-regularizing when $\hat{\Delta}(1,2)$ is sufficiently large.

One approach that could inform the search for a better regularized estimate would be a more thorough investigation of the implications of simultaneously obtaining regularized estimates of several dependent variables. In choosing our regularized estimator, we drew inspiration from the single superimposed pair-of-Gaussians example. However, the differences in the effect of STAR*D treatments at different levels of QIDS are dependent, even though we computed the $\tilde{\nu}_b^*$ estimator at each level of QIDS separately. In one sense this approach is appealing, since it does not rely on any knowledge of the underlying approximation space and could have been used equally well with a non-linear or even non-parametric regressor. On the other hand, using knowledge of the behaviour of the underlying regression algorithm could provide a way of pooling evidence at different levels of QIDS, for example, to aid in making finer distinctions between treatments.

Regardless, we have shown that any way forward in assessing the stability of optimal action choices must be cognisant of the potential problems caused by irregularity, and must take steps to avoid false confidence induced by these problems.

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. Collins, L., Murphy, S., Strecher, V.: The Multiphase Optimization Strategy (MOST) and the Sequential Multiple Assignment Randomized Trial (SMART): New methods for more potent e-health interventions. American Journal of Preventive Medicine **32**(5S) (2007) S112–118
2. Ernst, D., Geurts, P., Wehenkel, L.: Tree-based batch mode reinforcement learning. Journal of Machine Learning Research **6** (2005) 503–556
3. Laber, E., Murphy, S.: Small sample inference for generalization error in classification using the CUD bound. In: Proceedings of Uncertainty in Artificial Intelligence. (2008) 357–365
4. Margineantu, D.D., Dietterich, T.G.: Bootstrap methods for cost-sensitive evaluation of classifiers. In: Proc. 17th ICML. (2000) 582–590
5. Sutton, R.S., Barto, A.G.: Reinforcement Learning: An Introduction. MIT Press (1998)
6. Efron, B.: Bootstrap methods: Another look at the jackknife. Ann. Statist. **7**(1) (1979) 1–26
7. Efron, B., Tibshirani, R.: The problem of regions. Ann. Statist. **26**(5) (1998) 1687–1718
8. Chakraborty, B., Strecher, V., Murphy, S.A.: Inference for nonregular parameters in optimal dynamic treatment regimes. Statistical Methods in Medical Research, special issue on Clinical Trials in Mental Health (2008) To appear.
9. Wasserman, L.: All of Statistics: A Concise Course in Statistical Inference. Springer (2004)
10. Rush, A.J., et al., M.F.: Sequenced treatment alternatives to relieve depression (STAR*D): rationale and design. Controlled Clinical Trials **25**(1) (Feb 2004) 119–42
11. Schafer, J.L.: Analysis of Incomplete Multivariate Data. CRC Press (1997)