Quantifying Uncertainty in Batch Personalized Sequential Decision Making

Vukosi Marivate*, Jessica Chemali, Emma Brunskill**, Michael Littman†**
*Department of Computer Science, Rutgers University
**Department of Computer Science, Carnegie Mellon University
†Department of Computer Science, Brown University
vukosi@cs.rutgers.edu, jessica.chemali@gmail.com, ebrunskill@cs.cmu.edu, mlittman@cs.brown.edu

Abstract

As the amount of data collected from individuals increases, there are more opportunities to use it to offer personalized experiences (*e.g.*, using electronic health records to offer personalized treatments). We advocate applying techniques from batch reinforcement learning to predict the range of effectiveness that policies might have for individuals. We identify three sources of uncertainty and present a method that addresses all of them. It handles the uncertainty caused by *population mismatch* by modeling the data as a latent mixture of different subpopulations of individuals, it explicitly quantifies *data sparsity* by accounting for the limited data available about the underlying models, and incorporates *intrinsic stochasticity* to yield estimated percentile ranges of the effectiveness of a policy for a particular new individual. Using this approach, we highlight some interesting variability in policy effectiveness amongst HIV patients given a prior patient treatment dataset. Our approach highlights the potential benefit of taking into account individual variability and data limitations when performing batch policy evaluation for new individuals.

Introduction

Domains like medicine, education, and marketing involve sequential interventions with individuals: treating patients, teaching students and advertising to consumers. In these high-stake domains, there is a huge need and opportunity to create personalized predictions of the effectiveness of these interventions. Such information could be crucial to inform which strategy/treatment regime/policy to employ for a given individual.

Creating such personalized predictions of a policy using prior sequential intervention data (*e.g.*, previous treatment regimes) is challenging because such estimates should account for a number of different sources of uncertainty that arise from the following factors:

1. *Data sparsity*. If there is a limited amount of data that is relevant to the policy of interest, it will yield significant uncertainty in the resulting policy-return estimates.
2. *Population mismatch*. When the population of individuals in the observed dataset is quite different from the new individual for which we want to evaluate a policy, the resulting predictions will be uncertain.
3. *Intrinsic stochasticity*. Inherent uncertainty in resulting outcomes, even if the true probabilities of each outcome are precisely known, also makes a significant contribution to the uncertainty over the result of executing a policy.

There has been limited attention in the batch RL literature to personalizing the uncertainty evaluation over the predicted effectiveness of policies given past data. Algorithms like LSPI (Lagoudakis and Parr 2003) estimate the expected return of a new policy, but do not quantify uncertainty over the resulting estimate. In the online model-based RL literature, Strehl and Littman (2005) estimate model-based intervals that guide learning but do not evaluate policies. Mannor et al. (2007) quantify uncertainty of a policy by estimating the bias and variance of the *expected* value of a policy using estimated discrete transition and reward functions. Shortreed et al. (2011) introduced an approach to learn the optimal policy estimate and its uncertainty in continuous states from clinical trials (not observational data). Though some of these approaches address uncertainty due to data sparsity, they do not incorporate stochasticity uncertainty or population mismatch uncertainty. In our high stakes domains, it is important to take all three sources of uncertainty into account to create personalized policy-evaluation estimates.

In this work, we introduce an approach that produces personalized assessments of an input policy's effectiveness given batch data. To our knowledge, our model-based reinforcement-learning algorithm is the first approach that addresses all three important sources of uncertainty over predictive payoff. It handles *population mismatch* by modeling the data as a latent mixture of subpopulations of individuals instead of one homogeneous population; it explicitly quantifies *data sparsity* by accounting for the uncertainty in the estimated model parameters; and it incorporates *intrinsic stochasticity* by computing percentile ranges of the effectiveness of a policy instead of only the policy's expected effectiveness. Using this approach we highlight interesting variability in policy effectiveness amongst HIV patients given a prior patient treatment dataset. Our approach highlights the potential benefit of taking into account individual variability and data limitations when performing batch policy evaluation for new individuals.

Problem Setup

We are interested in predicting a personalized *range* of returns when executing a particular policy, taking into account the three sources of uncertainty discussed earlier. To do so, we extend the classic MDP model in two key ways: We introduce *latent class MDPs* to capture subpopulations that vary in their underlying dynamics models; and an *interval loss function* to capture the inherent stochasticity in policy outcomes.

To define a latent class MDP, we augment the standard model, $\langle S, A, P, R, \gamma\rangle$, with a set of classes C. The transition function becomes conditioned on the class: $P_c(s' \mid a, s)$ is the probability that state s will transition to state s' given action a for agents in class $c \in C$. In addition, these latent classes are partially observable via a set of d-dimensional feature vectors (e.g., demographic features), one for each class. We assume that each feature vector, of d independent variables, is drawn from a multivariate normal with mean $\boldsymbol{\mu}_c$ and diagonal covariance matrix $\mathbf{\Sigma}_c$. Implicitly, we assume that there will be a small finite number of latent classes. We test this assumption later on when exploring an HIV dataset.

In the learning problem, we assume that we are given data in the form of observational samples, $O = \{\zeta_1, \ldots, \zeta_N\}$. Each sample ζ_i has an associated feature vector and a trajectory of T_i steps. The i-th sample x_i can be written as

$$\zeta_i = \{f_i\}\{(s_i^1, a_i^1, r_i^1), \ldots, (s_i^{T_i}, a_i^{T_i}, r_i^{T_i})\},$$

where f_i is the d-dimensional feature vector. The reward r_i^t is assumed to be a function of the state or state-action pair. We define the returns that we receive for an observation sample ζ as $\Re(\zeta) = \sum_{t=1}^{T_i} \gamma^{t-1} r_t$.

We also introduce an $\alpha \in (0, 1]$ interval loss function. The α interval loss function captures the uncertainty over the returns of a policy for a particular individual in the class. We define this loss function with a lower range bound ℓ and the upper range bound u, as

$$J(u, \ell|\zeta, \alpha) = \begin{cases} \frac{\alpha}{2}(u - \ell), & \text{if} \ell \leq \Re(\zeta) \leq u \\ \Re(\zeta) - \frac{\alpha}{2}\ell - (1 - \frac{\alpha}{2})u, & \text{if } \Re(\zeta) > u \\ \frac{\alpha}{2}u + (1 - \frac{\alpha}{2})\ell - \Re(\zeta), & \text{if } \Re(\zeta) < \ell \end{cases} \tag{1}$$

The penalties are defined so that the expected loss is minimized by setting the bounds so that, on average, a $\alpha/2$ fraction of the observations fall below the interval and a $\alpha/2$ fraction of the observations fall above the interval.

Indeed, let

$$\begin{aligned} \mathbb{E}[J(u, \ell|\mathbf{O}, \alpha)] &= \frac{\alpha}{2}\int_{\ell}^{u} (u - l)p(R|\mathbf{O})dR \\ &+ \int_{-\infty}^{\ell} [\frac{\alpha}{2}u + (1 - \frac{\alpha}{2})\ell - R]p(R|\mathbf{O})dR \\ &+ \int_{u}^{\infty} [-\frac{\alpha}{2}\ell - (1 - \frac{\alpha}{2})u + R]p(R|\mathbf{O})dR. \end{aligned} \tag{2}$$

It is minimized by the roots $\ell^\star$ and $u^\star$ of $\frac{\partial \mathbb{E}[J]}{\partial \ell}$ and $\frac{\partial \mathbb{E}[J]}{\partial u}$, respectively. It is easy to show that they are solutions to:

$$P(R < \ell^\star) = \frac{\alpha}{2} \quad \text{and} \quad P(R > u^\star) = \frac{\alpha}{2}. \tag{3}$$

Modeling Uncertainty in Latent Class MDPs

In this section, we present how to estimate our augmented MDP from a set of observational data. Then, we discuss how to quantify the different types of uncertainty to produce personalized ranges of a policy's returns.

Finding a Latent Class MDP

We present an expectation-maximization algorithm for estimating the parameters, $\Psi_M = (\rho_1, \cdots, \rho_M, \mu_1, \cdots, \mu_M, \Sigma_1, \cdots, \Sigma_M, P_1, \cdots, P_M)$, of a latent class MDP with $C = \{1, \ldots, M\}$ (a priori unknown) classes, given observational data. We name this approach the Latent Class Search (LCS) algorithm. Our goal is to maximize the likelihood of observing a set of N observational samples $O = \{\zeta_1, \ldots, \zeta_N\}$ (*i.e.*, agent features and associated trajectories). The probability of observing features f_i, associated with ζ_i, given that they are drawn from model c is $L_{\mathcal{N}}(\zeta_i|c) = \mathcal{N}(f_i|\mu_c, \Sigma_c)$. Similarly, the probability of observing a transition to state s^{t+1} from s^t after action a^t given that it is drawn from class c is $P_c(s^{t+1} \mid a^t, s^t)$ for all $s \in S$, $a \in A$ and $c \in C$. Therefore, the probability of observing the trajectory $(s_i^1, a_i^1), \ldots (s_i^T, a_i^T)$ associated with ζ_i given that it is drawn from class c is:

$$L_T(\zeta_i|m) = \prod_{t=1}^{T} P_c(s_i^{t+1} \mid a_i^t, s_i^t).$$

Defining ρ_{ic} as the prior probability of assigning observation i to class c, the log likelihood of our samples is

$$L(\mathbf{O}|\Psi_M) = \log \prod_{i=1}^{N} \sum_{c \in C} \rho_{ic} L_{\mathcal{N}}(\zeta_i|\mu_c, \Sigma_c) L_T(\zeta_i|P_c). \tag{4}$$

From these building blocks, the LCS EM algorithm is:

1. In the **E-step** of the h-th iteration of the EM, calculate
$$\tau_{ic} = \frac{L_{\mathcal{N}}(\zeta_i|\mu_c\Sigma_c)L_T(\zeta_i|P_c)\rho_{ic}}{\sum_{c' \in C} L_{\mathcal{N}}(\zeta_i|\mu_{c'}, \Sigma_{c'})L_T(\zeta_i|P_{c'})\rho_{ic'}} \tag{5}$$
as the probability that observation i belongs to class c.
2. In the **M-step**, iterate $\Psi_M^h \rightarrow \Psi_M^{h+1}$ by finding the maximum likelihood estimates for the Gaussian and multinomial mixture models separately[1]. Since we can split the likelihood function into two, the EM algorithm uses the standard EM maximization step for the Gaussian mixture models representing features (Bilmes and others 1998) as well as that for the multinomial mixture model representing the transition functions.

Additionally, for each latent class c, calculate the maximum likelihood estimate of the reward function $\hat{R}_c(s, a, s')$ for all $s, s' \in S$, $a \in A$ and $c \in C$. This estimate helps quantify uncertainty.

We do not know the number of models M a priori so we run EM with several values of M and pick the best one using cross-validation (Smyth 2000): We vary the number of latent

[1] We omit the equations due to space limitations.

classes and evaluate the likelihood of the validation set. We choose the M for which the likelihood of the validation data is no longer increasing.

Computing an Interval of Possible Returns

The output of the LCS algorithm is a latent class MDP with M classes as well as membership probabilities of each training trajectory to each of the latent classes. We use these classes in two ways to estimate the return ranges.

First, we explicitly represent the uncertainty we have over the latent class MDP parameters due to the limited data[2] by not treating the resulting estimates from the EM procedure as point estimates, but instead creating a Bayesian posterior probability over the latent class MDPs. We take all the trajectories and use their soft assignments to each class to produce a Dirichlet posterior distribution over the multinomial transition model probabilities associated with each latent class MDP. In more detail, for each class, for each state–action pair (s, a), we define $Dir(\lambda_{(s,a)})$, where $\lambda_{(s,a)}$ is a count vector for each next state s'. This vector is set as the number of occurrences of (s, a, s') triples experienced in the training observation data, weighted by the probability that each trajectory was assigned to model c.

We also want to handle the population mismatch uncertainty that can occur for a new individual. In particular, we do not know which latent class the individual falls into. As such, for a new individual i, we can calculate the probability that they are associated with latent class c using only their features f_i. The algorithm calculates the value of an augmented Equation 5 that functions without trajectories:

$$w_{ic} = \frac{\mathcal{N}(f_i|\mu_c \Sigma_c)\rho_{ic}}{\sum_{c' \in C} \mathcal{N}(f_i|\mu_{c'}, \Sigma_{c'})\rho_{ic'}}, \tag{6}$$

and $\sum_{c \in C} w_{ic} = 1$. We can expand Equation 6 as we get partial transition information from the individual.

We now describe how we generate a range of the policy returns for a new individual i. We first compute w_{ic} and then repeat the following procedure many times: We first sample a latent class c given the individual's probability weight vector $\boldsymbol{w}_i$. We then sample a transition model for each state–action pair $\bar{P}_c$ from class c's associated Dirichlet distributions. We then perform a trajectory rollout, using the policy of interest to select actions for the states encountered, and the sampled transition model to generate the simulated transitions. We record the resulting reward obtained during this rollout, and then repeat this whole process. We then report an upper and lower range of the resulting values, as described in the loss function (1). We refer to the complete algorithm, LCS + Interval Estimation, as the Latent Structure and Uncertainty (LSU) algorithm.

Personalized Treatment Uncertainty

In this section, we detail the application of our approach to a real-life observational collection. We used an HIV (Human Immunodeficiency Virus) dataset from the EUResist project (Zazzi et al. 2012). This data differs from that of clinical trials in that it is an amalgamation of observational datasets from different patients, hospitals, and European countries. Our experiment is not a rigorous evaluation of the efficacy of HIV treatments or EUResist, but serves to illustrate the potential use of our approach with actual observational data from important areas such as medicine.

EuResist Dataset

The EuResist dataset consists of 18467 patients undergoing HIV treatment therapies. The dataset was previously used to build models that predict whether a drug would be effective or not for a patient. Current literature in the area predicts the response between 4–12 weeks. For a drug to be classified as being effective, it must reduce the viral load 100-fold from baseline or result in the virus being undetectable. We used the dataset to evaluate the effectiveness of sequential therapies, similar to the approach of Shortreed et al. (2011). A significant difference is that we used observational data instead of clinical trial data. Our approach potentially makes it possible to bring a larger amount of data to bear on policy evaluation.

In the analysis, we considered the patients who underwent at most 2 different treatment therapies over a 24-month period. The periods took place between January 2000 and December 2010. The viral load is the state variable and we tracked its changes monthly over 24 months. We interpolated the viral loads for months in which there was no data. Similar to Shortreed et al. (2011), we further encoded patient's treatment stage. We labeled their first treatment therapy as Stage 0 with any further treatment switches numbered consecutively (*i.e.*, $s_{i,j}$ is state i in Stage j).

The patients' continuous features in our latent class MDP were: baseline viral load, baseline CD4 count, baseline CD4 percentage, age and number of previous treatments.[3] The features were standardized via a linear re-scaling of the features so that each of the features had zero mean and standard deviation of 1. With assistance from HIV health-care experts, we identified the top 10 therapy/drug cocktail groups occurring in the reduced data set, discarding data that used therapies outside these groups. Each unique therapy was taken as an action. The state space was discretized by binning the values of the viral load. The bins for the viral load, in copies/mL, were [0.0,50,100,1K,100K]. State $s_{0,0}$ and state $s_{0,4}$ are thus viral loads between 0 and 50.0 copies/mL and 1K and 100K copies/mL, respectively. The reward function was the negation of the Area Under Curve (AUC), calculated monthly, of the viral load over the period being studied. This reward function favors a patient having a lower viral load over a long period of time. We calculated the return with $\gamma = 1.0$ but with a maximum of 24 steps (24 months).

[2] We do not model uncertainty due to the local EM search, but we can later assess how well our approach performs in terms of capturing real individuals' returns.

[3] The dataset includes virus genomic information for only some of the patients, so we did not use it as a feature. Including genomic information is a great opportunity for future work as it is a valuable marker for resistance and mutations.

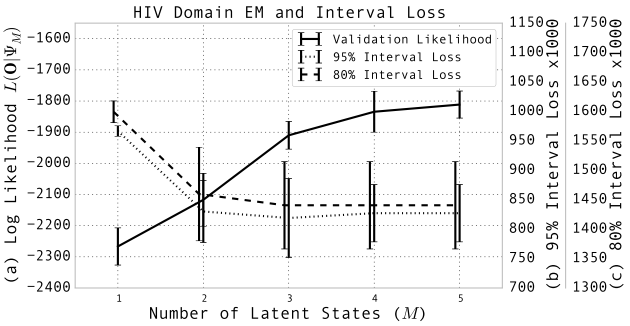

Figure 1: Likelihood and loss functions for the LSU algorithm with mixed observational data from the HIV dataset.

Discovering Latent Classes

An initial step to validating our approach is to determine if there is indeed latent structure. To find the number of latent classes into which the HIV patient data should be split, we again randomly partitioned the data into a training set (for EM) and validation set. The training set, after standardization and removing outliers, consisted of 6552 samples while the validation set had 200 samples. To calculate the interval loss function for the dataset, we sought out the most common two-stage treatment policies that were observed in the dataset. From the subpopulation that followed these top policies, we sampled 50 patients and made them part of the validation set of 200. We plot the results of running our LSU algorithm on the observational HIV data in Figure 1.

The likelihood of observing the data increases as we increase the number of latent classes. Similarly, the α-interval loss (a measure of the uncertainty returns of a policy) drops, indicating that our interval estimates improve as well (at least up to $M = 3$). The plot indicates that the HIV data indeed has latent structure—there appear to be at least 3 different subpopulations of patients in the data. We use interval loss instead of likelihood to choose the number of latent classes because it is a more direct measure of prediction quality.

With $M = 3$, we ran an additional experiment to uncover what the LSU algorithm does when presented with individuals from different latent classes. We drew a subsample of 200 patients from the original dataset and used their features to compute individual latent class membership probabilities (6). We plot the probability of a patient falling into the dominant latent class, the class that has the highest estimated prior probability, vs. the predicted returns of a policy.[4] As our algorithm computes ranges, we present the 80% ranges of returns and do this for two different treatment policies. The interval range estimates are shown in Figure 2. The results have several noteworthy properties. First, for patients with a high probability of falling into the dominant latent class, policy π_2 seems to have a distinct advantage. Not only does it result in higher overall returns, but the ranges of returns are more tightly clustered. Second, for patients who clearly are not in the dominant class, probability close to zero, the opposite is true: π_2 now has extremely large ranges of possible returns, whereas π_1 seems to do quite well and have fairly tight ranges. There is substantially more uncertainty over the return of π_2 for patients who do not fall into the dominant class. The source of this high uncertainty is partly due to a lack of data for the other 2 latent classes as compared to the dominant class, resulting in wider variability in the sampled transition functions. Finally, note that this plot provides evidence that ignoring latent classes results in less accurate evaluation predictions. In particular, a single class model will either evaluate π_1 or π_2 as superior. In fact, each is superior, but under different circumstances.

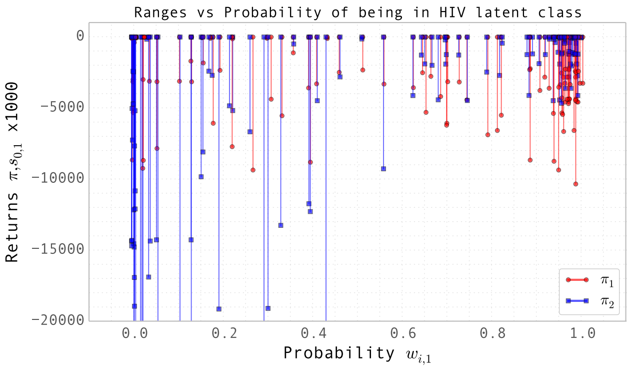

Figure 2: LSU intervals for 2 HIV therapy polices vs. probability of an individual being in dominant latent class

There are still further opportunities to delve into datasets such as this one, but our modest goal here was to show that the LSU algorithm can find latent classes that improve predictions.

Conclusion

We presented a method for using batch observational data to provide personalized estimates of the effectiveness of policies. The approach first searches for latent structure, and then quantifies uncertainty within and across the latent classes to compute effectiveness ranges for new individuals. We also show that there is an opportunity to apply this approach to critical domains such as healthcare, where data is becoming more plentiful.

Acknowledgments

We would like to thank Dr. Benjamin Muthambi (Pennsylvania Department of Health) and Dr. Tonya Crook (Penn State Hershey Infectious Diseases) for assistance with interpreting disease data and ongoing discussions about modeling HIV treatments.

References

Bilmes, J. A., et al. 1998. A gentle tutorial of the EM algorithm and its application to parameter estimation for

[4] To improve comparability, the graph includes only patients that had a baseline viral load of between 50–100 copies/mL (start state of $s_{0,1}$). The latent class visualized has an estimated prior probability of 0.62, while the two other classes have 0.24 and 0.14.

Gaussian mixture and hidden Markov models. *International Computer Science Institute* 4(510):126.

Lagoudakis, M. G., and Parr, R. 2003. Least-squares policy iteration. *The Journal of Machine Learning Research* 4:1107–1149.

Mannor, S.; Simester, D.; Sun, P.; and Tsitsiklis, J. N. 2007. Bias and variance approximation in value function estimates. *Management Science* 53(2):308–322.

Shortreed, S.; Laber, E.; Lizotte, D.; Stroup, T.; Pineau, J.; and Murphy, S. 2011. Informing sequential clinical decision-making through reinforcement learning: an empirical study. *Machine Learning* 84(1):109–136.

Smyth, P. 2000. Model selection for probabilistic clustering using cross-validated likelihood. *Statistics and Computing* 10(1):63–72.

Strehl, A. L., and Littman, M. L. 2005. A theoretical analysis of model-based interval estimation. In *Proceedings of the 22nd International Conference on Machine Learning*, 856–863.

Zazzi, M.; Incardona, F.; Rosen-Zvi, M.; Prosperi, M.; Lengauer, T.; Altmann, A.; Sonnerborg, A.; Lavee, T.; Schülter, E.; and Kaiser, R. 2012. Predicting response to antiretroviral treatment by machine learning: The EuResist project. *Intervirology* 55(2):123–127.