Published in final edited form as:

Med Decis Making. 2016 November ; 36(8): 927–940. doi:10.1177/0272989X15605091.

Some Health States Are Better than Others: Using Health State Rank Order to Improve Probabilistic Analyses

Jeremy D. Goldhaber-Fiebert, PhD¹ and Hawre J. Jalal, MD, PhD^{2,1}

¹Stanford Health Policy, Centers for Health Policy and Primary Care and Outcomes Research, Department of Medicine, Stanford University, Stanford, CA, USA

²Health Policy and Management, University of Pittsburgh, Pittsburgh, PA, USA

Abstract

Background—Probabilistic sensitivity analyses (PSA) may lead policymakers to take nonoptimal actions due to misestimates of decision uncertainty caused by ignoring correlations. We developed a method to establish joint uncertainty distributions of quality-of-life (QoL) weights exploiting ordinal preferences over health states.

Methods—Our method takes as inputs independent, univariate marginal distributions for each QoL weight and a preference ordering. It establishes a correlation matrix between QoL weights intended to preserve the ordering. It samples QoL weight values from their distributions, ordering them with the correlation matrix. It calculates the proportion of samples violating the ordering, iteratively adjusting the correlation matrix until this proportion is below an arbitrarily small threshold. We compare our method to the uncorrelated method and other methods for preserving rank ordering in terms of violation proportions and fidelity to the specified marginal distributions along with PSA and Expected Value of Partial Perfect Information (EVPPI) estimates, using two models: 1) a decision tree with 2 decision alternatives; 2) a chronic hepatitis C virus (HCV) Markov model with 3 alternatives.

Results—All methods make trade-offs between violating preference orderings and altering marginal distributions. For both models, our method simultaneously performed best, with largest performance advantages when distributions reflected wider uncertainty. For PSA, larger changes to the marginal distributions induced by existing methods resulted in differing conclusions about which strategy was most likely optimal. For EVPPI, both preference order violations and altered marginal distributions caused existing methods to misestimate the maximum value of seeking additional information, sometimes concluding that there was no value.

Conclusions—Analysts can characterize the joint uncertainty in QoL weights to improve PSA and Value-of-Information estimates using open-source implementations of our method.

Corresponding Author: Jeremy D. Goldhaber-Fiebert, PhD, 117 Encina Commons, Stanford, CA 94305-6019, jeremygf@stanford.edu, Phone: 650-721-2486, Fax: 650-723-1919.

Disclaimer: The findings and conclusions in this report are those of the authors and do not necessarily represent the official position of the Department of Veterans Affairs, or the U.S. Government.

Previous Presentations: An earlier version of this work was presented at the 2014 SMDM North American meeting.

Keywords

Probabilistic Sensitivity Analysis; Joint Distribution; Parameter Correlation; Value of Information; Expected Value of Partial Perfect Information; Bias; Correlated Parameters

Introduction

Standard assessments of decision uncertainty may lead risk-averse policymakers to take nonoptimal actions because they imply incorrectly high or low levels of certainty about model-based recommendations.¹ This results from two important types of information derived from decision-analytic models and how this information is generated. First, models provide estimates of expected health benefits and costs for each alternative to identify the most cost-effective for implementation.² Second, through probabilistic sensitivity analyses (PSA), models assess how uncertain this recommendation is and gauge the expected value of obtaining additional information (VOI).^{3–6} While accurate point estimates of model inputs permit accurate cost-effectiveness assessments, estimates of decision uncertainty may still be biased depending on how the PSA and VOI are conducted.^{3,7}

Although PSAs are mostly governed by standard practices,^{3,8} important methodological issues remain. Specifically, estimates of decision uncertainty from PSAs are only as good as the assumptions made about the joint uncertainty of a model's inputs.^{9–12} In practice, PSAs typically assume independence among the inputs' uncertainty distributions, enabling the use of univariate, parametric distributions.⁸ In some instances, correlations between the uncertainty distributions of specific model inputs – mainly related to disease natural history or test characteristics – can be obtained via methods like empirical model calibration.^{13–18} However, without substantial primary data, methods to establish the joint uncertainty distributions of other major categories of model inputs (e.g., treatment efficacies, side-effects, and adherence, costs, QALY weights, etc.) remain challenging, a situation that previous authors have acknowledged.^{3,10,14,16,17,19–23}

Our study develops a practical method for establishing the joint uncertainty distribution of QoL weights for a set of health states. Our method exploits the fact that while one may only have point estimates and confidence intervals to infer the uncertainty describing the QoL weight of each health state, additional information regarding their joint uncertainty distribution can be recovered by using the ordinal preferences over these health states. Without such correlations in QoL weight uncertainties, more severe health states like cancer may be valued as being better than near-perfect health – a situation whose falsehood is a certainty.

Methods

We describe a method for using health state preference orderings to reduce the bias in PSAs and VOI analyses. First, we review existing approaches and assess them with respect to ideal properties that such methods should have. We then describe our method and compare the performance of each method using both a simple, illustrative decision model and a more complex, previously published Markov model.²⁴

Problem Definition

The joint uncertainty of QoL weights for health states should preserve the natural preference ordering over those states:

$$\text{Healthy} \succ \text{Mild} \succ \text{Moderate} \succ \text{Severe} \succ \text{Dead}$$

implying:

$$u(\text{Healthy}) \succ u(\text{Mild}) \succ u(\text{Moderate}) \succ u(\text{Severe}) \succ u(\text{Dead})$$

furthermore:

$$\begin{aligned} u(\text{Healthy}) &= 1 \\ u(\text{Dead}) &= 0 \end{aligned}$$

Sampling from independent univariate uncertainty distributions for each state can produce some samples that violate the preference ordering. For example, if we index the samples of a PSA with subscript i , then it is possible that $u_i(\text{Mild}) < u_i(\text{Moderate})$ for some samples. Violations are most common when distributions overlap (i.e., closer means, larger standard errors, or both). Violations can bias PSA estimates of how likely a strategy is to be optimal and the estimated VOI for resolving uncertainty about QoL weights.

Existing Methods

In situations where individuals' QoL weight assessments for multiple health states are not available, a variety of approaches exist to establish the joint uncertainty distribution for QoL weights. Each makes trade-offs between achieving various properties, which we would argue an ideal method would simultaneously satisfy:

1. preserving the rank order of health states
2. preserving the marginal (univariate) distributions for each QoL weight consistent with those in the literature
3. practicality of being simple to define and implement

Table 1 details existing methods that have been applied or proposed and compares them to the ideal properties listed above.^{3,25,26} None achieves all ideal properties simultaneously.

New Method (Induced-Correlations)

The core concept of our approach relies on reducing preference order violations by correlating QoL weights. However, sampling correlated QoL weights from a joint probability distribution can be extremely difficult. Thus, we first take a large number of independent samples (e.g., 10,000) from the QoL weights' independent marginal distributions – one vector of QoL weight samples per health state. Then, we “induce” correlation among these vectors by sorting them using the algorithm described below. Such correlation results in reducing the number of preference order violations without altering the

marginal uncertainty distributions. We build up an explanation of our full algorithm step by step. First we provide intuition about how our sorting algorithm works with a simple numerical example with only two vectors of sampled QoL weights for two health states. Then, we show how this approach can be extended to handle vectors of sampled QoL weights for more than two health states for which we may have more complex preference orderings.

Our first example focuses on two uncertain QoL weights for two hypothetical health states: “mild complications” which is rationally preferred over “moderate complications”. We assume that the utility of mild complication is distributed as a $\text{beta}(9,1)$ and utility of moderate complication is distributed as $\text{beta}(8,2)$. The mean utility of mild (0.9) is greater than the mean of utility of moderate (0.8) consistent with our preference order. However, if we treat the uncertainty distributions as independent, it cannot be guaranteed that for all random samples from them, the utility of mild will be greater than moderate.

The columns of **Step 1** of Table 2 present an example with 10 independent random samples from each distribution as is common in standard PSAs. Both the second and the eighth samples (rows) violate the preference order. **Step 1** of Table 2 also shows the rank of the sample within each column vector of QoL weights that play a crucial role in our algorithm. Because of the small sample size in the example, by chance the two utility vectors are correlated at -0.45 . Obviously, for a very large sample, the QoL weights and their ranks would be uncorrelated.

To induce correlation between these two vectors and thereby eliminate preference order violations, we sort their values according to a multivariate standard normal distribution. The latter distribution acts as a *reference distribution* because it is relatively easy to sample from and its correlation structure is easy to specify compared to sampling from other distributions such as correlated “bi-beta” distributions. **Step 2** of Table 2 shows 10 random samples taken from a bivariate standard normal distribution with a correlation of 0.9, which represents the correlation we would like to induce between the two QoL weight vectors in Step 1.

We are only interested in the rank ordering of the values from the multivariate normal distribution, not the values themselves. This is because we only use these ranks to sort our original QoL weight samples in **Step 1** of Table 2 to induce correlation and remove preference order violations as shown in **Step 3** of Table 2.

The original values in **Step 1** of Table 2 are simply rearranged within their respective columns to produce the same rank ordering as the corresponding columns sampled from the correlated multivariate normal distribution (guaranteeing the same rank (Spearman) correlation for both). Notice that by rearranging the values as shown in **Step 3** of Table 2, we introduce correlation and avoid preference order violations. Also notice that since we have only sorted the values, we have not changed the marginal distributions in any way. Finally, in the illustrative example, our decision to set the correlation at 0.9 is entirely arbitrary. If we increase the correlation to 1.0, we can guarantee the minimum number of preference order violations, but in doing so we suggest the uncertainty of one QoL weight is entirely explained by the other which may not always be the case.

Instead of choosing correlation arbitrarily, we seek to induce a correlation level that balances the uncertainty in the QoL weights and the number of samples that violate the preference ordering. The flowchart in Figure 1 describes an iterative technique of obtaining such correlation. Our starting point in our iterative algorithm is near-perfect correlation between the utility weights because it minimizes the number of violations. First we compute the percent of samples violating the preference ordering at this level of correlation (ρ). Then, we incrementally reduce the correlation by a small fixed amount (δ). This likely increases the percent of violations. We do this until the percent of violations reaches our tolerance threshold (ϵ) at which point we stop, accepting the last correlation level producing preference order violations below ϵ .

Extending the Sorting Algorithm to Three or More QoL Weights with Complex Preference Orderings

Extending the algorithm above to three or more vectors involves two additional challenges. First, we may not always have strict preference orderings for every pair of health states. As a result, the preference ordering among the various health states may be more complex than a simple ranking of these states. Thus, sorting multiple utility weights involves obtaining a potentially full correlation matrix. Second, the correlation matrix obtained must have certain mathematical properties that allow multiple utility vectors to be simultaneously sorted. (Below we illustrate a technique to approximate this matrix. In addition, we provide open-source implementations of our algorithm in both R and MATLAB.)

Step 1—First, we create three matrices: (1) a preference order matrix (Q) which defines the preference ordering between each pair of health states; (2) an independent QoL weight matrix (U) whose columns are formed by taking random samples from the independent marginal distribution of each QoL weight; (3) a reference matrix (R) which has the same dimensions as U , but is sampled from an equal number of independent standard normal distributions. We use Q , U and R in Step 2 to construct a matrix U^* that preserves the independent marginal distributions and has an arbitrarily small number of preference order violations.

1a) Q is an $s \times s$ matrix, where s is the number of health states. Q permits full flexibility to specify pairwise preferences between health states. The user can express preference orderings for each $\{j,k\}$ pair of health states, where $j = \{2, \dots, s\}$ and $k = \{1, \dots, j-1\}$. Thus, forms a lower triangular matrix, such that:

$$Q = \begin{bmatrix} 0 & 0 & \cdots & 0 & 0 \\ q_{2,1} & 0 & \cdots & 0 & 0 \\ q_{3,1} & q_{3,2} & \cdots & 0 & 0 \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ q_{s,1} & q_{s,2} & \cdots & q_{s,s-1} & 0 \end{bmatrix},$$

where each $q_{j,k}$ takes a value to signify that

$$q_{j,k} = \begin{cases} 1 & j^{th} \text{ state is preferred over } k^{th} \text{ state} \\ -1 & k^{th} \text{ state is preferred over } j^{th} \text{ state} \\ 0 & \text{there is no strict preference order} \end{cases}$$

See Appendix A for an example constructing a reference matrix Q .

1b) The utility matrix (U) is an $n \times s$ matrix. Similar to a standard PSA with uncorrelated parameters, this matrix is formed by taking n arbitrary samples from the appropriate independent univariate uncertainty distributions for the QoL weights of each health state. Each column vector of U contains n random QoL weight samples for a particular health state. The columns are clearly uncorrelated. In Step 2, we induce correlation among them by sorting them until the preference order violations in U reaches a predefined tolerable threshold, if such threshold is attainable.

1c) Before starting the iterative process, we obtain a reference matrix (R). The dimension of R is $n \times s$, and is formed by taking n random samples from s independent standard normal distributions. We will use R for sorting vectors of U in the steps below.

Step 2—We use an iterative process to calculate the desired correlation matrix (C) that when used to correlate the vectors of U , will transform it towards achieving the ideal properties described above. If we allow perfect correlations among all utility vectors (i.e., all elements of C set equal to 1), we guarantee the minimum possible preference order violations without altering the marginal distributions. However, with perfect correlation, one can specify uncertainty for only one QoL weight upon which all other QoL weights depend. Therefore, the goal of this step is to find the minimum correlation needed for each pair of QoL weight vectors to preserve the rank ordering set by Q while allowing uncertainty in all QoL weights.

The iterative process consists of two loops. An outer loop iterates over all pairs of health states, and an inner loop reduces the correlation coefficient (ρ) between the selected pair until a tolerable level of violations (ϵ) is reached. We record this correlation, switch to the next pair in the outer loop, and continue until all pairwise correlations are determined.

2a) Outer loop: Let the pair $\{j, k\}$ represent the selected pair of QoL weight vectors in the outer loop. We set the matrix X equal to the j^{th} and k^{th} column vectors of U , and the matrix Y equal to the j^{th} and k^{th} column vectors of R .

2b) Inner loop: In this step we induce ρ correlation between the two vectors in X . We start with near-perfect correlation (i.e., $1 - \delta$), where δ is a small positive quantity (e.g., 0.01). Then, we iteratively decrement ρ by δ until the percent violations reaches the tolerance level (ϵ).

Let Σ represent a correlation matrix:

$$\Sigma = \begin{bmatrix} 1 & \rho \\ \rho & 1 \end{bmatrix}.$$

Since Σ is positive and definite, we can express $\Sigma = PP'$, where P is the lower triangular matrix with the same dimension as Σ computed using the Cholesky decomposition. We compute a new matrix $Y^* = YP'$. The resulting matrix Y^* has the desired correlation (ρ). Then, we sort X 's columns independently to have the same rank ordering as Y^* by reordering X 's entries based on the rank orders in Y^* as described in Iman et al.²⁷ Thus, the columns in the resulting matrix X^* will have the same rank (Spearman) correlation coefficient as Y^* and preserve the marginal distributions of the j^{th} and k^{th} QoL weights.

For each inner loop iteration, we compute the percent of samples violating the preference ordering defined by $q_{j,k}$. If the percent of violations is less than ϵ , we decrement ρ by . We repeat step 2b until the sample violations reach our pre-specified threshold ϵ or ρ equals 0. We then record the correlation $c_{j,k} = \rho$ in matrix C , and move to the next pair of health states in the outer loop (2a). After finishing all pairwise combinations, we proceed with Step 3.

Step 3—After calculating all desired pairwise correlations in C , we sort the vectors in U and obtain the matrix U^* which approaches the goal of avoiding preference order violations while preserving the marginal univariate uncertainty distributions of all the QoL weight vectors combined. This process closely follows Step 2b. However, unlike Σ , we cannot always use C directly as it is not guaranteed to be positive and definite (required to be used as a correlation matrix) and hence its Cholesky decomposition may not be feasible. Therefore, we first compute the eigenvalues (B) and eigenvectors (V) for C , set any of eigenvalues > 0 in B equal to a very small positive number (e.g., 0.0001) to produce B^* , and compute the adjusted correlation matrix $C^* = VB^*V^{-1}$. Again, we express C^* as MM' , using the Cholesky decomposition to compute M . Then, we sort the columns in the reference matrix (R) and compute $R^* = RM'$. Finally, we sort the columns in U independently to have the same rank as the columns in R^* .²⁷ The sorted utility matrix U^* minimizes preference order violations while maximizing the joint uncertainty among the QoL weights, and approaches the other desired properties. Since this step may impose a new ordering of the values within the QoL weight vectors, analysts are encouraged to check the percent sample violations before and after this step to ensure that the algorithm meets the specified limits and if not to adjust tolerance thresholds, distributional assumptions or both before rerunning the algorithm. In practice, the analyst must choose a value for ϵ , the proportion of samples allowed to violate the preference ordering. The choice can be arbitrarily small and using 1% or 0.1% could be quite reasonable in many applications, though for smaller values of ϵ , the analyst must start with a larger number of PSA samples. Furthermore in situations where the analyst feels that the marginal distributions are much more uncertain than the preference ordering, it is possible to use a very small ϵ and then in a post-processing step after using our algorithm, drop any samples violating the preference order with minimal changes to the marginal given that such a small fraction of samples are dropped. Our approach may prove more efficient than rejection sampling methods used

alone, especially for complex preference ordering and computationally expensive models, given that it may reject fewer candidate samples.

Example Models Used for Methods Comparison

We compare our method to Uncorrelated Distributions, Rejection Sampling, Altering Marginal Distributions, and Cross-Sorting (Table 1). We focus on these methods because they do not fix correlation at an arbitrary assumed level, as the underlying correlation structure is generally highly uncertain. We compare these methods using two decision-analytic models described below.

Example Model 1—The first is a simple decision tree for a hypothetical disease with health states Mild and Moderate complications. The decision is between a less expensive and less effective “Drug A” and “Drug B” (see Appendix B for further details). While

Mild Complication $\succ$ Severe Complication

we are uncertain about the QoL weights for Mild and Moderate complications. We explore potential bias in PSA and VOI depending on the method used to form the joint uncertainty distribution for QoL weights under three prior information scenarios: High Uncertainty, Moderate Uncertainty, and Low Uncertainty, in which the expected values of the QoL weights remain the same but the standard errors are progressively smaller (see Appendix Table B1). Figure 2 illustrates the relationship of uncertainty and correlation on preference order violations for the QoL weights of Mild and Moderate. At low correlation and high uncertainty the proportions of violations (points above the orange line) are the highest. Increasing the correlation substantially reduces these violations, which is the core of our approach.

Example Model 2—The second is a previously published Markov model of chronic HCV infection.²⁴ Briefly, the model tracks individuals with chronic genotype 1 HCV monoinfection and with an initial level of liver fibrosis quantified in terms of Metavir score F0–F4. Liver fibrosis can progress towards advanced liver disease including decompensated cirrhosis, hepatocellular carcinoma, and liver transplantation. More advanced liver disease lowers QoL and increases mortality. Treatment leading to sustained virologic response (SVR) arrests liver progression, improves QoL and survival, and reduces medical costs. In our example, the model involves uncertainty about the QoL weights for 9 health states (defined in terms of fibrosis and SVR), a preference ordering in which some health states are not known to be unambiguously better than others (e.g., whether SVR after F4 fibrosis is better or worse than F2 fibrosis without SVR). The model involves a decision between three treatment alternatives: no treatment, dual therapy with pegylated interferon and ribavirin, and triple therapy involving boceprevir along with pegylated interferon and ribavirin. While newer HCV treatments are available, we do not include them to aid in simplicity of exposition.

Assessing Performance

We compare each method's performance with each example model using the following procedure. We start by comparing methods simultaneously using two performance metrics: a) the proportion of samples violating preference orderings; b) changes to the marginal distributions. We compute the first metric by determining the average percent violations over all pairs of health states. For the second metric, we use the Kolmogorov-Smirnov (K-S) statistic to quantify, for each health state, the absolute difference between the Empirical Cumulative Density Function (ECDF) of the pre-specified marginal distribution and the ECDF implied by the samples produced by each method. The K-S statistic is equal to zero if the distribution does not change, and equal to one if the method changes the distribution. For both metrics, we subtract the values computed from 1 to generate a measure of fidelity [0, 1] where higher values imply greater fidelity.

We then evaluate how each method's performance on these two metrics changes with different amounts of uncertainty reflected in the marginal distributions (i.e., the 3 uncertainty scenarios listed above for Example Model 1 (see Appendix Table B1)).

Finally, we evaluate how each method's fidelity to preference ordering and the marginal distributions altered its PSA estimates (i.e., the likelihood of a given strategy being cost-effective at a given willingness-to-pay threshold and the expected incremental net monetary benefit (EINMB) of a strategy and the uncertainty of the EINMB estimate) and expected value of partial perfect information (EVPPI) estimates (i.e., the value one could expect to gain by eliminating uncertainty about a parameter). For PSA, we considered two measures because prior research has noted that the first measure of likelihood of a strategy being optimal does not capture how much better a strategy is when it is optimal.²⁸ We hypothesized that both preference order violations and changes to marginal uncertainty distributions from the various methods would bias the PSA and EVPPI results in complex ways. For both the PSA and EVPPI, we use 10,000 samples and to ensure that comparisons of methods were not due to randomness we used the same random seed to draw PSA samples to which each method is applied.

Making the Approach Accessible

To support external evaluation of our method and its wider use in decision models and cost-effectiveness analyses, we implemented versions for Matlab (Mathworks, Natick, MA) and R (The R Consortium). Code is provided in Appendix C1 and C2. All implementations take as inputs the independent QoL matrix (U) and the preference rank ordering matrix (Q), and output the sorted matrix U^* of the PSA samples. The rows of U^* represent the PSA samples, and the columns represent the QoL weights for the health states. This facilitates import into programs like TreeAge Pro (TreeAge Software, Inc., Williamstown, MA) (see Appendix D for detailed instructions).

Results

Our method maintains high fidelity to the univariate marginal distributions for QoL weights while simultaneously producing few preference order violations for both Example Model 1

and Example Model 2 (Figure 3, Panels A and B). As expected, the existing methods either maintain fidelity to the marginal distributions at the expense of violating preference order (Uncorrelated Distributions) or else avoid violations at the expense of changing marginal distributions (Rejection Sampling, Altering Marginal Distributions, and Cross-Sorting). While our method performs better on all metrics used, existing methods, particularly Altered Margins, may actually perform worse than the fidelity metric suggests because Altered Margins can shift the tails of the distribution more than the middle in a way that the K-S statistic is less sensitive to.

Our method's performance advantages are larger when the uncertainty reflected in the univariate marginal distributions is higher (Figure 3, Panel C). While all methods evaluated had greater difficulty simultaneously maintaining fidelity to the marginal distributions and keeping violation rates low when uncertainty was high, the existing methods had substantially larger performance degradations in the presence of higher uncertainty.

For PSA, failure to maintain fidelity to marginal distributions altered the fraction of times a strategy was considered optimal and sometimes resulted in different conclusions about which strategy was most cost-effective. For Example Model 1 (at a willingness-to-pay (WTP) of \$50,000/QALY for illustration), all methods suggest that "Drug B" is preferred to "Drug A". However the percent of simulations in which B is preferred to A vary from 56% to 93%. In addition the Expected Increment Net Monetary Benefit (EINMB) also varies from \$146 to \$793, indicating that the results of PSA vary substantially depending on the method chosen to avoid preference order violations (Table 3, Upper Panel). The EINMB of "Drug B" is higher than "Drug A" under all methods, though our method and the Uncorrelated Distribution are closest to the INMB [computed by setting all parameters to their mean values (i.e., \$150)]. Likewise the standard deviation of the EINMB tends to be underestimated in the existing methods. For Example Model 2, the two methods that are faithful to the marginal distributions estimate a greater likelihood that "Triple Therapy with Boceprevir" is preferred (WTP=\$44,000/QALY for illustration) relative to the existing methods that suggest that Dual Therapy is most likely optimal (Table 3, Lower Panel). This result is driven by the fact that the existing methods tend to estimate an EINMB for Boceprevir that is lower (and more certainly so) than the methods that are faithful to the marginal distributions.

For EVPPI, violating preference ordering and altering the original marginal distributions resulted in existing methods generally biasing the estimates of value of seeking additional information. For Example Model 1, the Uncorrelated Distributions method, which permits a substantial proportion of preference order violations and implies that information gained about one parameter does not provide information about other parameters, overestimated EVPPI for all QoL weights (Table 4). In contrast, methods that reduced preference order violations at the expense of the fidelity to the marginal distributions suggested that there was lower but non-zero value to obtaining additional information about either QoL weight. Notably, the bias of existing methods when compared to that estimated with the Induced-Correlations method can go either way. For the multiple health states considered in Example Model 2 (Appendix E), EVPPI estimates differed with generally higher values for

Uncorrelated Distributions. While for other health states, existing methods produced lower EVPPI estimates than the Induced-Correlations method.

While the EVPPI values estimated in both examples were relatively small for all methods, they represent the value of obtaining information for each affected patient – the relevant quantity for assessing whether there is insufficient value to obtain additional information is the analogous population EVPPI. For example, if there are 100,000 patients (discounted to present net value) who could benefit from improving information and the cost of a study to obtain information were \$300,000, then in Example Model 1, one would correctly conclude that there is insufficient value to obtain this information only using the following methods: our method, Cross-Sorting, or Rejection Sampling. This is because the population EVPPI estimates for either health state for these methods is below \$300,000. A similar example with the same study cost but with 1,000,000 patients for Example Model 2 shows that whereas our method (along with Uncorrelated Distributions and Altering-Margins) would suggest the possibility that obtaining additional information may be sufficiently valuable, Cross-Sorting and Rejection Sampling incorrectly conclude that there is insufficient value to obtaining additional information (Appendix E).

Discussion

Probabilistic Sensitivity Analyses and Value of Information Analyses are powerful tools that are increasingly used and required by journals. Methods to ensure that they are free from bias are therefore of growing importance. While it is difficult in general to specify correct joint uncertainty distributions for all model parameters, for Quality of Life weights of health states, preference orderings over those states can inform their unbiased joint uncertainty distribution – e.g., it is certain that states like “Healthy” should have higher QoL weights than states like “Diseased” in all PSA and VOI samples.

We developed a method that achieves this goal without distorting the univariate marginal uncertainty distributions specified for the QoL weight of each state. The comparison of our method to existing alternative methods shows that ours performs best in terms of simultaneously avoiding preference order violations while remaining faithful to the marginal distributions. Further, we used two example models to show that these differences can matter for both PSA and VOI estimates. To ensure that the results of our analyses are mostly driven by variations in the methods and not simply due to randomness, we repeated the analysis of method performance for both Example Model 1 and Example Model 2 one hundred times using different random seeds (Appendix F). The variations in the results were very small over the PSA analyses, almost entirely attributable to the methods rather than to chance alone. To enable further assessment by researchers and ease adoption by analysts, we provide Open Source implementations of our method in Matlab and R.

Our approach has a number of other advantages. It is not limited by the number of model health states it can accommodate, by restrictions on the types of marginal uncertainty distribution used for QoL weights, or by what correlations patterns it can incorporate. It runs quickly because its Open Source implementation employs commonly used, high performance languages and because its algorithm relies on simple sorting, univariate

optimization, and closed-form solutions. In addition, the iterations in the outer loop could be executed on parallel computer processing threads to further increase efficiency. Finally, it is possible to employ our method along with linear regression meta-modeling or other emulator techniques to update previously run PSA analyses that use existing methods to establish their joint uncertainty distributions even without access to the original model as long as one has access to a set of PSA results (sampled parameter values to provide the univariate marginals and corresponding costs and effects for each intervention considered).^{29–32}

Our study contributes to existing work on specifying uncertainty distributions for PSA and VOI that has been highlighted by recent best practice guidelines.³ Influential work by Briggs has helped to define the rationale for standard parametric forms primarily for univariate marginal distributions for many parameter types.²⁰ For joint distributions, Ades has developed Markov Chain Monte Carlo (MCMC) estimation techniques¹⁰, though current performance and convergence challenges of MCMC may hinder its widespread adoption. In other fields, researchers use copula methods to fit joint distributions that are constrained by their marginal. However, these methods are best suited to cases when one wishes to recover a joint distribution based on points that have been generated by the true distribution.²¹ Primarily for joint distributions of natural history and intervention effectiveness parameters, health analysts use empirical calibration, though it is still not commonly practiced.^{14,16,22} One challenging case is of parameters describing behaviors that may be correlated (e.g., uncertainty distributions about average levels of condom use, needle sharing, seeking HIV testing, and adhering to ART treatment may be correlated). Recent work has begun to use empirical calibration for multiple risk behaviors (e.g., starting and quitting smoking).¹⁷ Techniques that use Bayesian updating or other approaches to characterize joint uncertainty of several parameters (e.g., the bivariate normal distribution) have been described for test sensitivity and specificity and for correlated treatment effect estimates.²³

Though the motivation for our study is the joint uncertainty of QoL weights, it could be applied more generally to other parameters where there is agreement about the ordering of variable values. For example, the joint uncertainty distribution of some cost or probability parameters could be characterized using our method.

Our method has limitations. First, similar to existing methods, parameters outside of these QoL weights are assumed independent. While we believe going from independent univariate to joint uncertainty for groups of parameters most likely reduces bias in PSA and VOI, it could potentially increase bias relative to estimates from independent univariate distributions. Future theoretical and numerical simulation work could shed light on this area. Second, our method is not the only way to maintain marginal distributions while respecting preference orderings, and hence, in some cases, alternative approaches may identify joint distributions that better reflect the uncertainty in combinations of parameters. Imagine a bivariate case in which the QoL weight of the less preferred state is on the y-axis and the more preferred state is on the x-axis. In addition to limiting violations in the northwest corner, our approach also reduces the number of samples that occupy the far southeast corner of the parameter space (see Figure 1). Although, other approaches could, in theory, produce joint distributions that occupy a greater portion of this southeast corner (or their

higher dimensional analogs), in numerical experiments, we have noted convergence challenges in fitting these types of joint distributions while respecting the marginal distributions. In addition, we believe that analysts may find limiting the samples in the far southeast corner desirable. This is simply because it limits extreme differences in QoL weights between the compared states by avoiding samples in which the better state is assigned an extremely high QoL weight while the less preferred state is assigned an extremely low QoL weight. While our study considered both a simple 2-state Markov model and a more complex published example showing in both that uncorrelated QoL weights could influence results, it will be important to evaluate how much such biases influence results across a range of real-world applications. Finally, in some applications, even with Open Source implementations available, analysts may reasonably consider the trade-off between simplicity of method and description versus accuracy of results when selecting the best method for their particular application.

Recent guidelines highlight the importance of ensuring that distributions used for PSA and VOI analyses reflect the true joint uncertainty of model parameters to avoid bias.³ We developed a practical method for capturing the joint uncertainty of QoL weights that reflects their specified marginal distributions while respecting the preference rank order of their corresponding health states; the method outperforms existing methods for addressing this challenge. Analysts should consider employing our method to reduce bias in PSA and VOI estimates given our findings and the availability of an Open Source implementation.

Acknowledgments

The authors wish to acknowledge helpful conversations with Karen Kuntz and Jay Bhattacharya on this study and related work that supports it.

Funding: Supported in part by US NIH/NIA Career Development Award (K01AG037593-01A1; PI: Dr. Goldhaber-Fiebert) and by the Department of Veterans Affairs for Dr. Jalal.

Appendices for Some Health States Are Better than Others: Using Health State Rank Order to Improve Probabilistic Sensitivity Analyses

Appendix A: Example of a Preference Order Matrix (Q)

The matrix below illustrates the preference order matrix we used for our HCV model example (Example Model 2). The 9 health states are listed on the rows and columns, and the cells indicate the specified pairwise preference orderings. Each cell indicates if the row state is better than (1), indifferent to (0), or worse than (−1) the column state. This matrix provides more flexibility for real-life models than a simple vector of ranked health state because the matrix allows for the possibility that one could be indifferent for choices between some health states. For example, in the matrix below the preference of Sustained Virologic Response (SVR) after cirrhosis and the HCV infection with moderate fibrosis is allowed to share the same rank (i.e., the preference ordering allows indifference for choices between these states).

	HCV mild fibrosis (F0, F1)	SVR after mild fibrosis	HCV moderate fibrosis (F2, F3)	SVR after moderate fibrosis	Compensated cirrhosis (F4)	SVR after cirrhosis	Decompensated cirrhosis	HCC	Liver transplant (after)
HCV mild fibrosis (F0, F1)									
SVR after mild fibrosis	1								
HCV moderate fibrosis (F2, F3)	-1	-1							
SVR after moderate fibrosis	-1	-1	1						
Compensated cirrhosis (F4)	-1	-1	-1	-1					
SVR after cirrhosis	-1	-1	0	-1	1				
Decompensated cirrhosis	-1	-1	-1	-1	-1	-1			
HCC	-1	-1	-1	-1	-1	-1	0		
Liver transplant (after)	-1	-1	-1	-1	0	-1	1	1	
Row is better than column									
1 = better, -1 = worse, 0 = Preference ordering cannot be determined									

Appendix B: Details of Example Model 1

We use a simple decision tree to illustrate our approach (Figure B1). The model evaluates the net benefit of two hypothetical drugs (Drug A and Drug B) for treating a hypothetical disease (condition X). For simplicity, we assume that both drugs can either result in mild or moderate complications. However, Drug B has a lower probability of moderate complication compared to Drug A (0.5 and 0.6, respectively). Drug B is also more expensive than Drug A, \$200 vs. \$100, respectively. The mild and moderate utility weights are uncertain. The utility weight of mild is sampled from $\text{beta}(7,3)$, and the utility weight of moderate is sampled from $\text{beta}(6.5, 3.5)$. Thus, the expected utility of Drug A is $0.40 \times 0.70 + 0.60 \times 0.65 = 0.67$, and the expected utility of Drug B is $0.50 \times 0.70 + 0.50 \times 0.65 = 0.675$. The expected net monetary benefit of A is $0.67 \times \$50,000/\text{QALY} - \$100 = \$33,400$, and the expected net monetary benefit of B is $0.675 \times \$50\text{K}/\text{QALY} - \$200 = \$33,550$. Thus, the incremental net monetary benefit of B relative to A is \$150.

Figure B1.
Markov States for Example Model 1.

Table B1

Parameter Distributions of QoL Weights

QoL Weights for Health States	Mean	Distribution		
		Base-case	Higher uncertainty (lower precision)	Lower uncertainty (Higher precision)
Mild	0.7	Beta(7,3)	Beta(0.7, 0.3)	Beta(70, 30)
Moderate	0.65	Beta(6.5, 3.5)	Beta(0.65, 0.35)	Beta(65, 35)

Appendix C1: Matlab Code to Build Correlation between Variables

A. Save these functions in a file named correlateUtilitiesMatlab.m


```

%% Function 1: determining the optimal correlation structure
function Ustar = correlateUtilitiesMatlab(U, Q, epsilon, delta)
% Independently find rank correlation at <= epsilon, and then compute a
% positive definite square matrix from the correlation matrix and
% finally sample the betas from this matrix.
[n,s]=size(U); %utility matrix
C = zeros(s,s); %correlation matrix
R = normrnd(0,1,n,s); %Reference matrix
for j = 2:s
    for k = 1:(j-1)
        rho = 1;
        X = U(:,{j k});
        Y = R(:,{j k});
        viol = 0;
        while viol<epsilon && rho>=0
            rho = rho - delta;
            B = induceRankCorrelation(X, Y, rho);
            viol = mean((Q(j, k) * B(:,1)) < (Q(j,k) * B(:,2)));
        end
        C(j,k) = rho + delta;
    end
    disp(j)
    disp(k)
end
% reconstruct the correlation matrix
C = C + C'; %symmetrical Rho from the lower triangle matRank.
for j = 1:s
    C(j,j) = 1; % the diagonal of ones.
end
% approximate Rho using the eigenvalues and vectors
[V, B] = eig(C); %compute the eigenvalues and eigenvectors
B(B<=0) = 0.0001; %Remove the problematic eigenvalues
Cstar = V * B * inv(V); %Reconstruct Rho
% reverse-engineer computing the covariance matrix based on the rank
% correlation.
Ustar = induceRankCorrelation(U, R, Cstar);
end

%% Function 2: Inducing Rank Correlation.
function Xstar = induceRankCorrelation(X, Y, Sigma)
if length(Sigma) == 1 %if only rho between 2 params is provided, convert it to a 2x2
matrix
    Sigma = [1 Sigma; Sigma 1];
end
[n, s] = size(X); % n = no. of simulations, s = no. of parameters.
Xstar = zeros(n,s); % Sorted Data
Xsorted = zeros(n,s); % intermediate matrix
Yrank=zeros(n,s);
YrankInv=zeros(n,s);
F = chol(Sigma);
Ystar = Y * F;
for ii = 1:s %sort each vector independently!
    Xsorted(:,ii) = sort(X(:,ii)); % FIRST: Sort the data
    [~,Yrank(:,ii)]=sort(Ystar(:,ii)); %compute what should be where. value 5 in
position 2
    YrankInv(Yrank(:,ii),ii)=1:n; % mapping the ranks to the row indices
    Xstar(:,ii)=Xsorted(YrankInv(:,ii),ii); %sort Beta1 by the ranking of X1
end
end

```

- B.** Example running the Matlab code: First change the working directory by using the cd command to the folder that contains the functions above. Then, test the example below

```

n = 30; %number of PSA samples
uMild= betarnd(9,1,n,1); %utility of mild complication
uMod= betarnd(8,2,n,1); %utility of moderate complication
uSevere=betarnd(7,3,n,1); %utility of severe complication

U=[uMild uMod uSevere]; %stack the utilities into a single matrix
disp(U)
corr(U)
Q = [ 0 0 0; %Lower triangular matrix of ranking rules.
-1 0 0; %for example, matRank(row 2, column 1) = -1 indicates
-1 -1 0]; %uMod (row 2) < uMild (col. 1)

epsilon = 0.05; %our threshold of tolerance for violations for each pair of
comparisons
delta = 0.01; %decrement the correlation between each pair by delta until epsilon is
reached

Ustar = correlateUtilitiesMatlab(U, Q, epsilon, delta); % Xout is the sorted utility
matrix Xout
disp(Ustar)
corr(Ustar)

```

Appendix C2: R Code to Build Correlation between Variables

- A.** Run these two functions first

```

# Please reference the manuscript for mathematical notations and variable
descriptions.
CorrelateUtils <- function(U, Q, epsilon, delta){
  n <- nrow(U) #number of PSA samples
  s <- ncol(U) #number of states
  R <- matrix(rnorm(n*s,0,1), n, s) #the reference matrix.

  C <- matrix(0,s,s) #a place holder for the correlation matrix
  Viol <- matrix(0,s,s) #violations matrix.
  for (j in 2:s){ # {j,k} is the selected pair of state comparisons
    for (k in 1:(j-1)){
      rho <- 1 # bivariate correlations
      X = U[,c(j,k)] #selected columns of U
      Y = R[,c(j,k)] #selected columns of R
      viol <- 0
      while(viol<epsilon & rho>=0){ #if these conditions are met, continue
        rho <- rho - delta #reduce correlation
        Xstar = induceRankCorrelation(X, Y, rho) #correlated utilities.
        viol = mean((Q[j,k] - Xstar[,1]) < (Q[j,k] - Xstar[,2])) #compute violations
        between the col. vectors.
      }
      Viol[j,k] <- viol
      C[j,k] <- rho + delta #record the desired correlation.
    }
    print(j) #just to show the column indices.
    print(k)
  }

  #Fill in the other elements of C.
  C = C + t(C)
  for (j in 1:s){
    C[j,j] <- 1 # the diagonal of ones.
  }

  ## Eigenvectors and Eigenvalues correction of C
  eigenResults <- eigen(C)
  B <- eigenResults$values
  V <- eigenResults$vectors
  B[B<=0] <- 0.0001 #to make sure C is positive definite, set eigenvalues<=0 to a very
  small positive number
  Cstar <- V %*% diag(B) %*% solve(V) #reconstruct C
  Ustar <- induceRankCorrelation(U, R, Cstar) #similar to above, induce the correlation.
  return(Ustar)
}

## To induce Rank correlation: inputs X: QoL vectors, Y is the reference vectors, and
Sigma is the correlation matrix.
induceRankCorrelation <- function(X, Y, Sigma){
  if (length(Sigma)==1){ #if Sigma is a single value, convert it to a 2x2 matrix.
    Sigma <- matrix(c(1, Sigma,
                      Sigma, 1), 2, 2)
  }
  n <- nrow(X)
  s <- ncol(X)
  #Initialize matrices.
  Xsorted <- matrix(0, n, s)
  Yrank <- matrix(0, n, s)
  Xstar <- matrix(0, n, s)
  P <- chol(Sigma) #compute the upper triangular matrix
  Ystar <- Y %*% P #Sort the values in the reference vectors by multiplying by P
  cor(Ystar)
  for (j in 1:s){
    Xsorted[,j] <- sort(X[,j]) #Sort each variable
    Yrank[order(Ystar[,j]),j] <- seq(1:n) #Reverse sort
    Xstar[,j]=Xsorted[Yrank[,j],j] #sort Xsorted to have the same ranks as Ystar.
  }
  return(Xstar) #return the sorted vectors.
}

```

- B. Example running the R code:** The code below shows a simple example running the R-functions above. This code can be easily scaled to models with a larger number of parameters, more complicated rank ordering, and a larger number of PSA simulations. To illustrate the advantage of our approach, we take 30 random samples for the QoL weights for 3 health states (Mild, Moderate and Severe). We highlight the samples violating the rank ordering of Mild > Moderate > Severe in bold before applying our approach, and following applying the approach. Our approach substantially reduces the number of violations from 14 to only 4 out of 30 only by resorting these vectors (notice that the values in each vector are not changed). It is worth mentioning that the 4/30=13.3% which is more than ϵ because the threshold is set at each of the three pairs of comparisons (i.e, mild<moderate, mild<severe and moderate<severe). In addition, we only took 30 PSA samples. Of course, modelers can specify lower thresholds if desired.

```

> n <- 30 #take 30 PSA samples
> s <- 3 #for 3 QoL weight vectors (uMild, uModerate, uSevere)
>
> U <- matrix(0, n, s)
> U[,1] <- rbeta(n, 5, 1) #uMild
> U[,2] <- rbeta(n, 8, 2) #uModerate
> U[,3] <- rbeta(n, 7, 3) #uSevere
>
> print(U)
      [,1] [,2] [,3]
[1,] 0.9452848 0.9467637 0.7927654
[2,] 0.9718163 0.9046839 0.7197689
[3,] 0.7018250 0.9516178 0.9168062
[4,] 0.9362115 0.9731232 0.7395100
[5,] 0.9673224 0.7629962 0.9415141
[6,] 0.9921592 0.9447734 0.7149522
[7,] 0.7460918 0.7939078 0.7643064
[8,] 0.9045340 0.5615901 0.4557477
[9,] 0.8194824 0.9329944 0.7961332
[10,] 0.7358385 0.8389831 0.8953949
[11,] 0.9628988 0.9147148 0.7231575
[12,] 0.9903658 0.8518611 0.7677091
[13,] 0.8652975 0.8570696 0.5146338
[14,] 0.7020859 0.6875280 0.8530701
[15,] 0.6359395 0.8282562 0.5867984
[16,] 0.8721592 0.8735072 0.6343484
[17,] 0.9865319 0.7653211 0.1724524
[18,] 0.9488603 0.7278119 0.7738976
[19,] 0.7315226 0.6239743 0.6143854
[20,] 0.8075001 0.9270035 0.5732640
[21,] 0.9179732 0.8585923 0.7225691
[22,] 0.9540382 0.9334569 0.6990218
[23,] 0.9501306 0.8453730 0.8422533
[24,] 0.9897966 0.6917094 0.5803898
[25,] 0.9763609 0.6854557 0.3964580
[26,] 0.8464002 0.6337638 0.6226866
[27,] 0.8891882 0.8387405 0.6384119
[28,] 0.9314972 0.8737716 0.5710378
[29,] 0.9846627 0.7640251 0.8209196
[30,] 0.6501247 0.8680327 0.5486670
> cor(U) #correlation of U (before sorting)
      [,1] [,2] [,3]
[1,] 1.0000000 0.04479504 -0.1134451
[2,] 0.04479504 1.0000000 0.3118726
[3,] -0.11344512 0.31187262 1.0000000
> ## alternatively U can be loaded from an excel spreadsheet
> #install.packages("gdata")
> #install.packages("xlsx")
> #library(xlsx)
> #library(gdata)
> #setwd("C:\\") # Set the working directory
> # Udata <- read.xlsx("Utilities.xlsx", 1, header = FALSE) #Read model inputs
  ts (sheet 1)
> # U <- as.matrix(Udata) #convert inputs into a matrix U
> ##
> Q <- matrix(c(0, 0, 0, #preference order matrix.
+             -1, 0, 0,
+             -1, -1, 0), s, s, byrow = TRUE)
> epsilon <- 0.05 #tolerance threshold
> delta <- 0.01 #decrement correlation (rho)
> Ustar <- CorrelateUtils(U, Q, epsilon, delta) #Induce correlation, and return
  Ustar.
[1] 2
[1] 1
[1] 3
[1] 2
> print(Ustar) #print values of Ustar (notice they are the same as U, but sorted)
      [,1] [,2] [,3]
[1,] 0.9673224 0.8585923 0.6143854
[2,] 0.8075001 0.7653211 0.7225691
[3,] 0.7020859 0.6875280 0.6384119
[4,] 0.9718163 0.8680327 0.7197689
[5,] 0.8464002 0.8389831 0.7395100
[6,] 0.9488603 0.8282562 0.5710378
[7,] 0.9921592 0.9731232 0.8953949
[8,] 0.9903658 0.9316178 0.7961332
[9,] 0.8652975 0.7939078 0.6343484
[10,] 0.9865319 0.9447734 0.7677091
[11,] 0.7358385 0.6917094 0.7149522
[12,] 0.9540382 0.8735072 0.7643064
[13,] 0.9763609 0.9046839 0.6226866
[14,] 0.9501306 0.8570696 0.5867984
[15,] 0.9045340 0.9147148 0.8530701
[16,] 0.9452848 0.8518611 0.6990218
[17,] 0.7315226 0.6854557 0.1724524
[18,] 0.9314972 0.7640251 0.5486670
[19,] 0.7460918 0.7278119 0.5803898
[20,] 0.9846627 0.9329944 0.7927654
[21,] 0.9362115 0.9270035 0.9415141
[22,] 0.8194824 0.8387405 0.7738976
[23,] 0.8721592 0.7629962 0.5146338
[24,] 0.6501247 0.5615901 0.4557477
[25,] 0.9897966 0.9467637 0.8422533
[26,] 0.9179732 0.8453730 0.7231575
[27,] 0.7018250 0.6337638 0.3964580
[28,] 0.9628988 0.9334569 0.9168062
[29,] 0.6359395 0.6239743 0.5732640
[30,] 0.8891882 0.8737716 0.8209196
> cor(Ustar) #print the correlation.
      [,1] [,2] [,3]
[1,] 1.0000000 0.9242409 0.5700377
[2,] 0.9242409 1.0000000 0.7529506
[3,] 0.5700377 0.7529506 1.0000000

```

Appendix D: Instructions to Incorporate the Matrix of Sorted Utilities (U^*) in TreeAge Pro

In this section, we illustrate how to incorporate the matrix U^* into TreeAge Pro. The process is relatively straightforward and involves instructing TreeAge to read the utility values from a table instead of taking random samples from the distributions (e.g., beta distributions). First, we need to create a table (e.g., tblUtilities) and copy the values in U^* and paste them into tblUtilities. Advanced users may wish to automate this process by saving the output

from Matlab to a datasheet (e.g., an Excel file) and programming TreeAge to populate tblUtilities from this file or by establishing a data connection using TreeAge's built-in SQL functionality. The second step involves changing the definition of the utility variables to values from the corresponding column in tblUtilities. Here, we assume that the first column of tblUtilities includes the values of the utility of mild complication (uMild). The new definition of this variable may look like this:

$$uMild = tblUtilities[_sample - 1; 1]$$

where the first parameter is the row index and the second parameter is the column index. TreeAge starts the row index from 0. Therefore we subtract 1 from TreeAge's built in function _sample which starts from 1. For example, at the beginning of the simulation _sample = 1, and the PSA reads the value in the first row (row index = 0) and the first column (column = 1) from tblUtilities. In the 100th PSA iteration, _sample = 100, and uMild is assigned the value in the row index 99 and the first column. Similarly the other utilities can be pointed to the other columns in tblUtilities. For example, the values for uModerate may be stored in the second column of tblUtilities. Thus, this variable definition in TreeAge may look like this:

$$uModerate = tblUtilities[_sample - 1; 2]$$

Clearly the only difference here is the column number. Finally, the user must confirm that the number of PSA simulations do not exceed the number of rows in tblUtilities.

Appendix E: Comparison of EVPPI Estimates for Example Model 2 Using Each Method

Methods	EVPPI of QoL Weight for*								
	Mild Fibrosis	SVR w/ Mild	Moderate Fibrosis	SVR w/ Moderate	Cirrhosis	SVR w/ Cirrhosis	DS	HCC	LT
Uncorrelated Distributions	19.91	0	0	0	0	0.72	6.00	0	0
Altering-Margins	0.48	0	0	0	0	0	0	0	0
Cross-Sorting	0	0	0	0	0	0	0	0	0
Rejection-Sampling	0	0	0	0	0	0	0	0	0
Induced-Correlations	0.36	0.44	0	0	0	0	0	0	0

* Health state abbreviations: DS (Decompensated Cirrhosis); HCC (Hepatocellular Carcinoma); LT (Liver Transplant).

Appendix F: Comparison of PSA Results Using Each Method for Each Example Model 1

Summary of 100 independent PSA analyses. Each PSA analyses used 100,000 random samples for the model parameters in Example Models 1 and 2. To ensure variability of the results among the various PSA analyses, we started each PSA analysis at a different random

number seed. The tables below show the summary results of these analyses for the measures listed in Table 3 of the manuscript. The summary statistics shown in the table below are the mean and standard deviation (in parentheses) of these measures over the results of the 100 PSA analyses for Example Model 1 and 2, respectively. The reason we compute the standard deviation over these statistics is that we wish to examine the differences between methods relative to the differences due to random Monte-Carlo noise. The standard deviations of the measures above were very small indicating that repeating the PSA analyses did not alter our results – the methods induce substantial differences beyond those due to chance alone.

Example Model 1

Methods	Probability Optimal*	Treatment B vs. Treatment A	
	Treatment B	EINMB	SD
Uncorrelated Distributions	56.11 (0.16)	149.98(3.01)	996.74 (2.05)
Altering-Margins	56.11 (0.16)	436.77(2.10)	661.61 (1.75)
Cross-Sorting	92.32 (0.08)	723.55(1.96)	614.60(1.47)
Rejection-Sampling	93.52(0.08)	795.11 (1.86)	640.02(1.49)
Induced-Correlations (Our method)	80.22 (0.50)	149.71 (2.95)	178.05 (0.70)

Example Model 2

Methods	Probability Optimal***			Boceprevir vs. No Treatment		Boceprevir vs. Dual Therapy	
	No Treatment	Dual Therapy	Boceprevir Therapy	EINMB	SD of EINMB	EINMB	SD of EINMB
Uncorrelated Distributions	28.15 (0.13)	32.70 (0.15)	39.15 (0.15)	3628.47 (30.59)	10239.0 2 (20.04)	-584.26 (10.81)	3529.75 (9.20)
Altering-Margins	28.87 (0.13)	34.46 (0.15)	36.67 (0.14)	3303.56 (30.67)	10167.93 (19.44)	-776.85 (10.72)	3471.58 (8.88)
Cross-Sorting	31.02 (0.13)	38.42 (0.14)	30.55 (0.14)	2396.76 (30.10)	9991.80 (19.57)	-1245.66 (10.17)	3376.10 (9.19)
Rejection-Sampling	30.23 (0.13)	36.36 (0.13)	33.41 (0.14)	2765.18 (30.46)	10086.0 8 (19.47)	-1033.83 (10.26)	3428.96 (9.33)
Induced-Correlations (Our method)	28.25 (0.13)	32.75 (0.14)	39.00 (0.15)	3592.07 (31.08)	10215.6 8 (19.39)	-602.73 (10.82)	3504.08 (8.93)

References

1. Claxton K. The irrelevance of inference: a decision-making approach to the stochastic evaluation of health care technologies. *J Health Econ* [Internet]. 1999 Jun; 18(3):341–64. Available from: <http://www.ncbi.nlm.nih.gov/pubmed/10537899>.
2. Claxton K, Cohen JT, Neumann PJ. When is evidence sufficient? *Health Aff* [Internet]. 2005; 24(1): 93–101. Available from: <http://www.ncbi.nlm.nih.gov/pubmed/23796298>.
3. Briggs A, Weinstein MC, Fenwick EL, Karnon J, Sculpher MJ, Paltiel D. Model parameter estimation and uncertainty analysis: A report of the ISPOR-SMDM Modeling Good Research Practices Task Force Working Group-6. *Med Decis Mak* [Internet]. 2012; 32(5):722–732. [cited 2012 Sep 19]. Available from: <http://www.ncbi.nlm.nih.gov/pubmed/22990087>.
4. Brennan, A.; Kharroubi, SA.; Chilcott, J.; O'Hagan, A. Presented at the 24th Annual Meeting of the Society for Medical Decision Making. Washington; 2002. A two level Monte Carlo approach to calculation expected value of sample information: how to value a research design.

5. Ades A, Lu G, Claxton K. Expected value of sample information calculations in medical decision modeling. *Med Decis Mak* [Internet]. 2004; 24(2):207–27. [cited 2012 Nov 18]. Available from: <http://www.ncbi.nlm.nih.gov/pubmed/15090106>.
6. Coyle D, Oakley J. Estimating the expected value of partial perfect information: a review of methods. *Eur J Heal Econ* [Internet]. 2008; 9(3):251–9. Available from: <http://www.ncbi.nlm.nih.gov/pubmed/17638032>.
7. Naveršnik K, Rojnik K. Handling input correlations in pharmacoeconomic models. *Value Health* [Internet]. 2012; 15(3):540–9. [cited 2014 Sep 5]. Available from: <http://www.ncbi.nlm.nih.gov/pubmed/22583465>.
8. Briggs A. Handling uncertainty in cost-effectiveness models. *Pharmacoeconomics* [Internet]. 2000; 17(5):479–500. [cited 2013 Mar 23]. Available from: <http://www.ncbi.nlm.nih.gov/pubmed/10977389>.
9. Smith AE, Ryan PB, Evans JS. The effect of neglecting correlations when propagating uncertainty and estimating the population distribution of risk. *Risk Anal* [Internet]. 1992; 12(4):467–74. Available from: <http://www.ncbi.nlm.nih.gov/pubmed/1480795>.
10. Ades A, Lu G. Correlations between parameters in risk models: estimation and propagation of uncertainty by Markov Chain Monte Carlo. *Risk Anal* [Internet]. 2003; 23(6):1165–1172. [cited 2014 Sep 14]. Available from: <http://onlinelibrary.wiley.com/doi/10.1111/j.0272-4332.2003.00386.x/full>.
11. Chessa AG, Dekker R, Van Vliet B, Steyerberg EW, Habbema JDF. Correlations in uncertainty analysis for medical decision making: An application to heart-valve replacement. 1999; 19(3): 276–286. *Med Decis Mak* [Internet]. [cited 2014 Sep 9]. Available from: <http://mdm.sagepub.com/cgi/doi/10.1177/0272989X9901900306>.
12. Clemen R, Winkler R. Limits for the precision and value of information from dependent sources. *Oper Res* [Internet]. 1985; 33(2):427–442. [cited 2014 Sep 14]. Available from: <http://pubsonline.informs.org/doi/abs/10.1287/opre.33.2.427>.
13. Caro JJ, Briggs AH, Siebert U, Kuntz KM. Modeling good research practices—overview: a report of the ISPOR-SMDM Modeling Good Research Practices Task Force-1. *Med Decis Making* [Internet]. 2012; 32(5):667–77. [cited 2014 Sep 9]. Available from: <http://www.ncbi.nlm.nih.gov/pubmed/22990082>.
14. Goldhaber-Fiebert JD, Stout NK, Ortendahl J, Kuntz KM, Goldie SJ, Salomon JA. Modeling human papillomavirus and cervical cancer in the United States for analyses of screening and vaccination. *Popul Health Metr* [Internet]. 2007; 5(11) [cited 2012 Jun 1]. Available from: <http://www.pubmedcentral.nih.gov/articlerender.fcgi?artid=2213637&tool=pmcentrez&rendertype=abstract>.
15. Salomon JA, Weinstein MC, Hammit JK, Goldie SJ. Cost-effectiveness of treatment for chronic hepatitis C infection in an evolving patient population. *JAMA* [Internet]. 2003; 290(2):228–37. Available from: <http://www.ncbi.nlm.nih.gov/pubmed/12851278>.
16. Salomon J, Weinstein M, Hammit JK, Goldie SJ. Empirically calibrated model of hepatitis C virus infection in the United States. *Am J Epidemiol* [Internet]. 2002; 156(8):761–773. [cited 2014 Sep 14]. Available from: <http://aje.oupjournals.org/cgi/doi/10.1093/aje/kwf100>.
17. Goldhaber-Fiebert, JD.; Brandeau, ML. Modeling and calibration for exposure to time-varying, modifiable risk factors: The example of smoking behavior in India. *Med Decis Mak* [Internet]. 2014 Jan 29. [cited 2014 Sep 9]; Available from: <http://www.ncbi.nlm.nih.gov/pubmed/24477078>
18. Kim JJ, Kuntz KM, Stout NK, Mahmud S, Villa LL, Franco EL, et al. Multiparameter calibration of a natural history model of cervical cancer. *Am J Epidemiol* [Internet]. 2007; 166(2):137–50. [cited 2012 Jul 5]. Available from: <http://www.ncbi.nlm.nih.gov/pubmed/17526866>.
19. Sculpher, M. The use of a probabilistic sensitivity analysis for decision making: the example of drug-eluting stents. 2004. [Internet] Available from: <http://www.york.ac.uk/che/pdf/teehtacosteff04.pdf>
20. Briggs, A.; Sculpher, M.; Claxton, K. *Decision Modelling for Health Economic Evaluation*. New York: Oxford University Press; 2006. [Internet][cited 2013 Mar 24]. Available from: <http://ideas.repec.org/b/oxp/obooks/9780198526629.html>

21. Clemen R, Reilly T. Correlations and copulas for decision and risk analysis. *Manage Sci* [Internet]. 1999; 45(2):208–224. [cited 2014 Sep 14]. Available from: <http://pubsonline.informs.org/doi/abs/10.1287/mnsc.45.2.208>.
22. Taylor DC, Pawar V, Kruzikas DT, Gilmore KE, Sanon M, Weinstein MC. Incorporating calibrated model parameters into sensitivity analyses: deterministic and probabilistic approaches. *Pharmacoeconomics* [Internet]. 2012; 30(2):119–26. Available from: <http://www.ncbi.nlm.nih.gov/pubmed/22149631>.
23. Hazen GB, Huang M. Large-sample Bayesian posterior distributions for probabilistic sensitivity analysis. *Med Decis Making* [Internet]. 2006; 26(5):512–34. [cited 2014 Jan 5]. Available from: <http://www.ncbi.nlm.nih.gov/pubmed/16997928>.
24. Liu S, Schwarzingler M, Carrat F, Goldhaber-Fiebert JD. Cost effectiveness of fibrosis assessment prior to treatment for chronic hepatitis C patients. *PLoS One* [Internet]. 2011; 6(12) [cited 2014 Sep 8]. Available from: <http://www.pubmedcentral.nih.gov/articlerender.fcgi?artid=3229483&tool=pmcentrez&rendertype=abstract>.
25. Jones M, Larsen P. Multivariate distributions with support above the diagonal. *Biometrika* [Internet]. 2004; 91(4):975–986. [cited 2014 Sep 5]. Available from: <http://biomet.oxfordjournals.org/content/91/4/975.short>.
26. Liu S, Cipriano L, Holodniy M, Owens DK, Goldhaber-Fiebert JD. New protease inhibitors for the treatment of chronic hepatitis CA cost-effectiveness analysis. *Ann Intern Med* [Internet]. 2012; 156:279–290. [cited 2014 Sep 20]. Available from: <http://annals.org/article.aspx?articleid=1132633>.
27. Iman R, Conover W. A distribution-free approach to inducing rank correlation among input variables. *Commun Stat Comput* [Internet]. 1982; 11(3):311–334. [cited 2014 Sep 14]. Available from: <http://www.tandfonline.com/doi/abs/10.1080/03610918208812265>.
28. Groot Koerkamp B, Hunink MGM, Stijnen T, Hammitt JK, Kuntz KM, Weinstein MC. Limitations of acceptability curves for presenting uncertainty in cost-effectiveness analysis. *Med Decis Mak*. 2007; 27(2):101–111.
29. Jalal H, Dowd B, Sainfort F, Kuntz KM. Linear regression metamodeling as a tool to summarize and present simulation model results. *Med Decis Making* [Internet]. 2013; 33(7):880–90. [cited 2013 Oct 13]. Available from: <http://www.ncbi.nlm.nih.gov/pubmed/23811758>.
30. Stevenson MD, Oakley J, Chilcott JB. Gaussian process modeling in conjunction with individual patient simulation modeling: a case study describing the calculation of cost-effectiveness ratios for the treatment of established osteoporosis. *Med Decis Making* [Internet]. 2004; 24(1):89–100. [cited 2012 Jan 23]. Available from: <http://www.ncbi.nlm.nih.gov/pubmed/15005958>.
31. Oakley JE, Hagan AO. Probabilistic sensitivity analysis of complex models: a Bayesian approach. 2004:751–769.
32. Fonseca DJ, Navarrese DO, Moynihan GP. Simulation metamodeling through artificial neural networks. *Eng Appl Artif Intell* [Internet]. 2003; 16(3):177–183. [cited 2014 Sep 9]. Available from: <http://linkinghub.elsevier.com/retrieve/pii/S0952197603000435>.

Figure 1.

Algorithm for the Simplified 2 Health State Version of the correlated distributions method. This algorithm shows the core concept in our approach which balances correlation between samples (?) with the percent of samples violating the preference order. The algorithm starts with near-perfect correlation and reduces this correlation by a pre-defined decrement (?) until the percent of samples violating the preference order reaches a tolerance threshold (?).

Figure 2.

Violations of health state preference order for different levels of correlation in the joint uncertainty distributions for QoL weights of mild and moderate complications. The joint distribution of QoL weights for the utility of being in the mild health state (x-axis) and in the moderate health state (y-axis) are shown for 10,000 random samples. The plots show the relationship of the percent of samples violating the preference rank ordering of $U(\text{mild}) > U(\text{moderate})$ for 5 levels of correlations between these two QoL weights and 3 levels of uncertainty in the distributions of the QoL weights (the percentages in the upper left corner of each subplot show the proportion of samples resulting in violations). As expected, increasing correlations, reducing uncertainty, or both decreases the proportions of samples resulting in violations. Utility of Mild is distributed as $\text{beta}(7,3)$, and utility of Moderate as $\text{beta}(6.5, 3.5)$ at moderate level of uncertainty. Notice that the mean of the distribution stays constant (0.7 and 0.65 for Mild and Moderate, respectively) at all levels of correlation and uncertainty.

Figure 3.

Performance comparison of our method (induced-correlations) to existing methods for capturing the joint uncertainty distribution of QoL weights. Panels A and B compare the performance of our approach (Induced-Correlations) to four other approaches involving avoiding preference rank order violations for Example Model 1 and Example Model 2, respectively. Performance involves simultaneous assessment on two different scales: (1) the percent of samples violating the preference rank order with each the method (x-axis), and (2) how much the CDFs of the QoL weight uncertainty distributions change with each method compared to the specified marginal distributions without correlation (y-axis). The optimal point (no rank violations and no difference in CDF) is marked with the arrow. Our approach does not change the marginal distribution (0 CDF violations) and induces approximately 5% sample violations as per the epsilon we set, which can be made arbitrarily small. Panel C shows the effect of increased uncertainty on these performance metrics for Example Model 1. As expected, the performance of all methods improves when uncertainty is low as shown by the proximity of all methods to the optimal point indicated by the arrow. Our approach

continues to perform relatively well even at higher uncertainty levels where existing methods' performance tends to degrade appreciably.

Author Manuscript

Author Manuscript

Author Manuscript

Author Manuscript

Table 1

Previously Used Methods and Their Achievement of Ideal Properties for Capturing Joint Uncertainty Distributions of QALY Weights

Method	Description	Achieves Ideal Properties		
		Preserve Ranks	Preserve Margins	Practical
Uncorrelated Distributions	Random samples from uncorrelated univariate distributions for the QoL weight of each health state; most commonly used in practice	N	Y	Y
Rejection Sampling	Implied Correlation: 0 Like Uncorrelated Distributions except that samples that violate preference order are discarded	Y	N	Y
Assumed Correlation	Implied Correlation: (0,1) Random samples from a joint distribution where a correlation structure is assumed based loosely on preference ordering but without explicit attention to violations or changes to marginal distributions (e.g., bivariate normal with correlation 0.3).	N	Y	Y
Fixed Increments	Implied Correlation: (0,1) as per assumption Random sample from the univariate distribution for the health state with the highest expected QoL weight. The QoL weights for the other health states are established by subtracting fixed increments ($g_1, g_2, \dots, g_{s-1}$) from this QoL weight.	Y	[†]	Y
Random Increments	Implied Correlation: 1 Like Fixed Increments except that the increments to be subtracted from the highest QoL weight are randomly sampled from uncertainty distributions for ($g_1, g_2, \dots, g_{s-1}$).	N	[‡]	[§]
Altering Marginal Distributions to Preserve Rank Order	Implied Correlation: (0,1) Random samples from uncorrelated univariate distributions for each QoL weight are treated as candidate QoL weights. The most preferred health state is assigned its candidate weight. Other health states are assigned the minimum of their candidate QoL weights and the candidate QoL weights of all more preferred health states.	Y	N	N
Cross-Sorting	Implied Correlation: (0,1) Based on a more general approach, ²⁵ the application to QoL weights involves sampling the univariate distribution of each QoL weight and then assigning the highest value to the most preferred health state, the next highest value to the next most preferred state, etc. Implied Correlation: (0,1)	Y	N	Y

[†]The marginal distribution of U(A) will be consistent, but unless the distributions of U(B) and U(C) have equal uncertainty to U(A) and just lower means, the method does not preserve their marginal distributions.

[‡]The marginal distribution of U(A) will be consistent, but unless the distributions of U(B) and U(C) have the same distributional shape to that of U(A) and just means that are *on average* lower and uncertainty that is a multiplicative factor of that of U(A), then the method does not preserve their marginal distributions.

[§]The method is relatively simple to define, explain, define, and implement in real-world applications if the distributions of g_1 and g_2 are simple and univariate. If they are complex and joint then this may not be the case.

Table 2

Illustration of Correlated Distributions Method Applied to QoL Weights for Two Health States

Step 1: Generate Uncorrelated QoL Weight Samples*				Step 2: Generate Samples from MVSN Distribution for Correlated Ranks				Step 3: Sort Step 1 Samples Using Correlated Rank Samples from Step 2*			
U (Mild)		U (Moderate)		To Rank U(Mild)		To Rank U(Moderate)		U(Mild)		U(Moderate)	
Sampled Value	Within-Column Rank**	Sampled Value	Within-Column Rank**	First Vector (x1)	Rank of x1 Within-Column	Second Vector (x2)	Rank of x2 Within-Column	Sampled Value	Within-Column Rank**	Sampled Value	Within-Column Rank**
0.970	8	0.899	8	0.457	7	0.208	7	0.958	7	0.897	7
0.627	1	0.908	9	−0.855	2	−0.785	2	0.862	2	0.733	2
0.922	6	0.897	7	−0.251	5	−0.735	3	0.917	5	0.735	3
0.958	7	0.767	4	−1.911	1	−0.940	1	0.627	1	0.558	1
0.917	5	0.735	3	1.669	10	1.698	10	0.993	10	0.919	10
0.992	9	0.832	5	1.444	9	0.665	8	0.992	9	0.899	8
0.898	3	0.733	2	−0.565	3	−0.388	6	0.898	3	0.890	6
0.862	2	0.919	10	1.009	8	0.847	9	0.970	8	0.908	9
0.993	10	0.558	1	0.259	6	−0.499	5	0.922	6	0.832	5
0.914	4	0.890	6	−0.344	4	−0.674	4	0.914	4	0.767	4

MVSN = Multivariate Standard Normal

* Preference order violations shown in bold. Such violations exist in Step 1 but are removed by the application of Step 2 to produce the samples in Step 3, which no longer has violations.

** Rank from smallest to largest

Table 3

Comparison of PSA Results Using Each Method for Each Example Model

Example Model 1						
Methods	Probability Optimal*		Treatment B vs. Treatment A			
	Treatment B		EINMB**	SD**		
Uncorrelated	55.87		145.70			
Distributions					1005	
Altering-Margins	55.87		433.94		667	
Cross-Sorting	92.28		722.19		620	
Rejection-Sampling	93.45		793.42		635	
Induced-Correlations (Our method)	80.80		152.02		180	

Example Model 2						
Methods	Probability Optimal***		Boceprevir vs. No Treatment		Boceprevir vs. Dual Therapy	
	Dual Therapy	Boceprevir Therapy	EINMB	SD of EINMB	EINMB	SD of EINMB
Uncorrelated	32.4	38.6	3,358	10,222	−650	3,555
Distributions						
Altering-Margins	34.7	35.9	3,018	10,132	−850	3,495
Cross-Sorting	38.6	29.7	2,113	9,974	−1,318	3,402
Rejection-Sampling	36.9	32.2	2,487	10,075	−1,104	3,458
Induced-Correlations (Our method)	33.2	38.1	3,304	10,180	−677	3,523

* Probability that A is Optimal is equal to 100 minus the number in this column.

** EINMB: Expected Incremental Net Monetary Benefit at example willingness to pay thresholds of \$50K/QALY and \$44K/QALY for the first and second examples, respectively; SD: standard deviation

*** Probability that “No Treatment” is Optimal is equal to 100 minus the sum of the numbers in the two columns below.

Table 4

Comparison of EVPPI Estimates for Example Model 1 Using Each Method

Methods	EVPPI of QoL Weight for	
	Mild	Moderate
Uncorrelated Distributions	\$217.00	\$215.00
Altering-Margins	\$28.00	\$5.00
Cross-Sorting	\$0.30	\$2.00
Rejection-Sampling	\$0.24	\$1.80
Induced-Correlations	\$0.00	\$0.00