

Techniques for Generic Probabilistic Inversion

D.Kurowicka¹, R.M. Cooke²

¹Department of Mathematics, Delft University of Technology, Delft, The Netherlands; d.kurowicka@its.tudelft.nl

²Department of Mathematics, Delft University of Technology, Delft, The Netherlands

Appeared in *Probabilistic Safety Assessment and Management* E.J. Bonano, A.L. Camp, M.J. Majors, R.A. Thompson (eds), Elsevier, 2002; 1543-1550.

ABSTRACT

Simple algorithms for probabilistic inversion are proposed, based on iterative sample re-weighting. One of these, iterative proportional fitting (*IPF*), dates from the 1930's and has been studied extensively with regard to contingency tables. A new algorithm is shown to converge even when *IPF* is infeasible.

KEYWORDS

Probabilistic inversion, iterative proportional fitting, maximal entropy, minimal information, I-projection.

1 Introduction

Probabilistic inversion arises in risk and uncertainty analysis when uncertainty distributions over model parameters (called *target variables*) are obtained via expert judgment. Experts are often unable to assess their uncertainty on these parameters directly, as there are no corresponding experiments on which to base their judgements. Instead, experts assess other, empirically measurable quantities (called *query variables*), which may, under suitable assumptions, be expressed as functions of the model parameters. For examples from consequence modelling, see (Kraan and Cooke [1], and (Harper *et al* [6]). In such cases distributions over the target variables must be pulled back onto the parameter space of the model in question. Alternatively, we say that the the model must be probabilistically inverted. The problem can be described mathematically as follows. Given a joint distribution for $\mathbf{Y} = [Y_1, Y_2, \dots, Y_n]$ and functions $G_i : \mathbf{R}^m \rightarrow \mathbf{R}, i = 1, 2, \dots, n$, find joint distributions for $\mathbf{X} = [X_1, X_2, \dots, X_m]$ such that $\mathbf{G}(\mathbf{X}) = [G_1(\mathbf{X}), G_2(\mathbf{X}), \dots, G_n(\mathbf{X})] \sim \mathbf{Y}$; where \sim means *has the same distribution as*. Note that this problem may be infeasible or if it is feasible it may have more than one solution. Therefore, we must have some method of selecting a preferred

distribution in case of non-uniqueness and some method of selecting a best fitting distribution in case of non-existence.

A few algorithms are presented in literature to solve probabilistic inversion problems, namely: conditional sampling, PARFUM (PARAmeter Fitting for Uncertain Models) (Cooke [3]), Hora and Young algorithm (see Harper *et all* [6]) and PREJUDICE (Kraan and Cooke [2],[10]). In this paper we present two new algorithms for solving probabilistic inversion problems based on sample re-weighting. The advantage of these algorithms is that they are dumb i.e. do not require intelligent steering. This makes them suitable for generic uncertainty analysis packages. The disadvantage is that they may be slow.

2 PARFUM

The PARFUM algorithm can be described in the following steps:

1. Choose a finite set $\chi \in \mathbf{R}^m$
2. Conditionalize the distribution of Y_i on $G_i(\chi)$, calling its mass function Q_i ;
3. Define the minimally informative “pull back” distribution on χ whose “push forward” distribution on $G_i(\chi)$ agrees with Q_i , that is, for $\mathbf{X} \in \chi$

$$P_i(\mathbf{X}) = \frac{Q_i(G_i(\mathbf{X}))}{\#G_i^{-1}G_i(\mathbf{X})};$$

4. Find a distribution P on χ which “best fits” the distributions P_1, P_2, \dots, P_n . The distribution which minimizes the relative information $\sum_{i=1}^n I(P_i, P)$ can be found, where

$$I(P_i|P) = \sum_{\mathbf{X} \in \chi} P_i(\mathbf{X}) \ln \left(\frac{P_i(\mathbf{X})}{P(\mathbf{X})} \right).$$

Let $S^k = \{P \in \mathbf{R}^k | P_i \geq 0, \sum_{i=1}^k P_i = 1\}$. It is not difficult to show ([3]):

Proposition 1 *Let $P^i \in S^k, i = 1, \dots, n$. Then $\min_{P \in S^k} \sum_{i=1}^n I(P^i|P) = \sum_{i=1}^n I(P^i|P^*)$ if and only if $P^* = (1/n) \sum_{i=1}^n P_i$.*

The advantage of this method is that it is always feasible and easily implemented. One disadvantage is that the conditional distributions Q_i might be different to those of Y_i , but this may be steered by the right choice of χ . More serious is the fact that the push forward of P need not have marginals that agree with the Y_i . This also can be influenced by steering, but is more difficult.

An alternative approach to parameter fitting for uncertain models was first developed by Hora and Young (see Harper et al [6]) and developed further by [10]). Whereas PARFUM solves an information minimization problem in the parameter space (step 4), these algorithms minimize information in the range of $Y_i, i = 1, \dots, n$. Unlike PARFUM, this minimization problem is not always feasible and the choice of χ is very critical. [10] apply duality theory to reduce infeasibility, eventually arriving at a feasible problem. While these algorithms produce good results for large problems, they require intelligent steering and are unsuitable for generic uncertainty analysis packages.

3 Sample Re-weighting

Sample re-weighting promises to provide generic methods for probabilistic inversion which do not require model inversion. The idea of re-weighting a sample to perform probabilistic inversion can be sketched roughly as follows. We generate a large sample from $X_1, \dots, X_n, G_1(X_1, \dots, X_n), \dots, G_m(X_1, \dots, X_n)$. Let the i -th sample be denoted $s_i \in \mathbf{R}^{n+m}$. Obviously each sample s_i has the same probability of occurring. If N samples have been drawn, then the sampling distribution assigning $p(s_i) = 1/N$ approximates the original distribution from which the samples were drawn. The idea is now to change the probabilities $p(s_i)$ so as to ensure that the distributions of G_1, \dots, G_m satisfy the user specified quantile constraints. No model inversions are performed, that is, we do not invert the function \mathbf{G} . To achieve reasonable results, the sample size N may have to be very large. This places strong restrictions on the type of methods which can be implemented in a generic uncertainty analysis program.

Notation Since the variables X_1, \dots, X_n play no role in choosing the weights $p(s_i)$, we may leave them out of the problem description. For notational convenience we put $Y_i = G_i(X_1, \dots, X_n); i = 1, \dots, m$.

A: M variables all with N samples are grouped in matrix Y . Hence $Y = [Y_1, Y_2, \dots, Y_M]$ where $Y_m = [Y_{1m}, Y_{2m}, \dots, Y_{Nm}]^T, i = 1, 2, \dots, M$. The matrix Y can be written

$$Y = \begin{bmatrix} Y_{11} & Y_{12} & \dots & Y_{1M} \\ Y_{21} & Y_{22} & \dots & Y_{2M} \\ \vdots & \vdots & \dots & \vdots \\ Y_{N1} & Y_{N2} & \dots & Y_{NM} \end{bmatrix};$$

B: We consider a vector $q = [q_1, q_2, \dots, q_K]$ of lengths of inter-quantile, or inter-percentiles intervals. For instance for $K = 4$, if we consider 5%, 50% and 95% quantiles then $q = [0.05, 0.45, 0.45, 0.05]$;

C: For $m = 1, \dots, M, j = 1, \dots, K - 1$ let r_{mj} be a number for which $P\{Y_m \leq r_{mj}\} = q_1 + \dots + q_j; r_{m,K} = \infty$. Define

$$A_1^m = \{i | Y_{mi} \in (-\infty, r_{m,1}]\}, \dots, A_j^m = \{i | Y_{mi} \in (r_{m,j-1}, r_{m,j}]\}, \dots, A_K^m = \{i | Y_{mK} \in (r_{m,K-1}, \infty)\}.$$

Note that $\mathcal{A}^m = \{A_j^m\}_{j=1}^K$ is a partition of $1, \dots, N$. We want the probability vector $[p(1), \dots, p(N)]$ to satisfy the *constraint set*³ \mathcal{C} :

$$\text{for all } k = 1, 2, \dots, K \text{ and all } m = 1, 2, \dots, M \\ \sum_{i=1}^N p_i \mathbf{I}_{A_k^m}(i) = q_k$$

where \mathbf{I}_A denotes the indicator function of a set A

The output of a re-weighting scheme is a vector $p = [p_1, p_2, \dots, p_N]$ of weights we should use such that after re-weighting samples, the constraints \mathcal{C} are satisfied 'as nearly as possible'.

³We note that these constraints do not really say that, e.g. r_{i1} is the q_1 -th quantile of Y_i . Indeed, the q_1 -th quantile is defined as the *least* number r_{ij} satisfying the above constraint. If Y_i is concentrated on a sparse number of points, then there may be many values $r_{i,j}$ satisfying the above constraints.

Optimization Approaches It is natural to approach the problem of probabilistic inversion as a constrained non linear optimization problem. The constraints \mathcal{C} are linear, and a convex objective function will have a unique solution, if the constraints are consistent. Minimal information is a natural choice of objective function. This yields the following optimization problem: Find $p = [p_1, p_2, \dots, p_N]$ such that p has minimum information with respect to the uniform distribution, under the constraints \mathcal{C} . Good performance for this problem has been obtained with an interior point solver, *if* the problem is feasible. If the problem is not feasible - and this is frequently the case - then several lines of attack suggest themselves ⁴.

Iterative algorithms to solve probabilistic inversion problem Iterative algorithms involve successively updating a probability vector $[p_1, \dots, p_N]$ so as to approach a solution satisfying constraints \mathcal{C} , or satisfying these constraints as closely as possible. We define the notion of adapting a probability measure to a partition and partition probability vector:

Definition 1 (Measure adapted to a partition; I-projection) Let $\mathcal{A} = \{A_j\}_{j=1}^K$ be a partition of $1, \dots, N$, $q = [q_1, \dots, q_K]$ be a probability vector, and let $p = [p_1, \dots, p_N]$ be a probability vector. Then

$$p_i^{\mathcal{A}} = \sum_{j=1}^K \mathbf{I}_{\{i \in A_j\}} \frac{p_i q_j}{p(A_j)} = \frac{p_i q_{j(i)}}{p(A_{j(i)})}$$

where $p(A_j) = \sum_{i \in A_j} p_i$ is the adaptation of p to $\{A_j\}_{j=1}^K$ with q , where $q_{j(i)}$ is defined by context. Let \mathcal{Q} be the set of all probability vectors which assign mass q_j to the set A_j , $j = 1, \dots, K$. Then $p^{\mathcal{A}}$ is called the I-projection of p on \mathcal{Q} .

The term "I-projection" is introduced by Csiszar ([4]), who shows that $p^{\mathcal{A}}$ is the unique element of \mathcal{Q} which minimizes $I(q|p)$, $q \in \mathcal{Q}$.

Iterative PARFUM This iterative algorithm simply repeats the PARFUM steps. If we have arrived at the probability vector p , we define the next iteration p' as:

$$p' = \frac{1}{M} \sum_{m=1}^M p^{\mathcal{A}^m}; \tag{1}$$

$$p'_i = p_i (1/M) \sum_{m=1}^M \frac{q_{k(i)}}{p(A_{k(i)}^m)}. \tag{2}$$

⁴We mention two. First, an objective function must be defined whose optimization will minimize infeasibility. For example, one could minimize the quantity Δ :

$$\Delta = \sum_{m=1}^M \sum_{k=1}^K \left[\sum_{i=1}^N p_i \mathbf{I}_{\{Y_{i,m} \in I_{k,m}\}} - q_k \right]^2$$

Of course this quantity has nothing to do with minimum information, and has nothing in particular to recommend it. Many other choices would be equally defensible. Solvers tested thus far have been unable to handle very large problems. Another approach would be to relax the constraints \mathcal{C} by replacing equality with 'equality up to ϵ '. In this case the choice of ϵ will be driving the solution, and this choice will be largely heuristic. Experience to date with infeasible problems has not been wholly satisfactory

Each measure p^{A^m} adapts the measure p so as to satisfy the quantile constraints for variable Y_m . From the discussion of PARFUM we see that p' is the probability vector which is 'closest' in the sense of relative information to all the measures p^{A^m} . If p satisfies the constraints \mathcal{C} , then $\frac{q_{k(i)}}{p(A_{k(i)}^m)} = 1$ and $p' = p$. In other words, the iterative PARFUM transformation has fixed points at all feasible probability vectors. The following theorem shows that this algorithm always converges, even when the constraints cannot be satisfied.

Theorem 1 *Let $\mathcal{Q}_1, \dots, \mathcal{Q}_M$ be closed convex subsets of $S^K = \{P \in \mathbf{R}^K | P_i \geq 0, \sum_{i=1}^K P_i = 1\}$. For $p^j \in S^K$, let q_i^j be the I-projection of p^j on \mathcal{Q}_i , $i = 1, \dots, M$. Let $p^{j+1} = (1/M) \sum_{i=1}^M q_i^j$. Then there exists $p \in S^K$ such that $\lim_{j \rightarrow \infty} p^j = p$.*

Proof Since \mathcal{Q}_i is closed and convex, q_i^j exists and is unique ([4], theorem 2.1). Define $F(p^j) = \sum_{i=1}^M I(q_i^j | p^j) \geq 0$. By proposition 1 and the fact that q_i^{j+1} is the I-projection of p^{j+1} on \mathcal{Q}_i , we have

$$\sum_{i=1}^M I(q_i^j | p^j) \geq \sum_{i=1}^M I(q_i^j | p^{j+1}) \geq \sum_{i=1}^M I(q_i^{j+1} | p^{j+1}). \quad (3)$$

Equality holds if and only if $p^j = p^{j+1}$. Thus, $F(p^j)$ is decreasing in j and converges. To show that p^j converges, pick $\varepsilon > 0$ and $j \in \mathbf{N}$ such that $F(p^j) - F(p^{j+1}) < \varepsilon$. Then $\sum_{i=1}^M I(q_i^j | p^j) - \sum_{i=1}^M I(q_i^j | p^{j+1}) < \varepsilon$. Writing this inequality element-wise:

$$\sum_{i=1}^M \sum_{k=1}^K q_{i,k}^j (\ln(q_{i,k}^j / p_k^j) - \ln(q_{i,k}^j / p_k^{j+1})) < \varepsilon.$$

Reversing the order of summation and substituting $\sum_{i=1}^M q_{i,k}^j = M p_k^{j+1}$, we find $MI(p^{j+1} | p^j) < \varepsilon$. The proof is concluded by applying the inequality ([9])

$$\sum_{k=1}^K |p_k^j - p_k^{j+1}| \leq \sqrt{2I(p^{j+1} | p^j)} \quad \square$$

The set \mathcal{Q}_i may be regarded as the set of vectors satisfying the i -th constraint; in particular, this may be the set of vectors satisfying the quantile constraints for Y_i . The middle term in (3) can be written as

$$G(q_1^j, \dots, q_M^j) = \sum_{i=1}^M I(q_i^j | p^{j+1}) = MH\left(\sum_{i=1}^M q_i^j / M\right) - \sum_{i=1}^M H(q_i^j);$$

where for $q \in S^K$, $H(q) = -\sum_{k=1}^K q_k \ln(q_k)$ is the *entropy* of q . If $q^i \neq q^j$ for some $i \neq j$ then $G(q_1, \dots, q_M) > 0$ by the strict convexity of H . The iterative PARFUM algorithm may be seen as minimizing the function G , and the value of G may be taken as a measure of 'how infeasible' the problem is. If $\bigcap_{i=1}^M \mathcal{Q}_i = \emptyset$, then the problem is infeasible. If $\bigcap_{i=1}^M \mathcal{Q}_i = \mathcal{Q} \neq \emptyset$ then the algorithm is observed to converge to an element of \mathcal{Q} ⁵ and $G(q_1, \dots, q_M) \rightarrow 0$. In this case the I-projection of the starting point p^1 on \mathcal{Q} exists and is not in general equal to the p given by the theorem. A sufficient condition for feasibility is given in the following:

Theorem 2 *Let $p_{ij\dots n}$ denote $p(\mathcal{A}_i^1 \cap \mathcal{A}_j^2 \dots \cap \mathcal{A}_n^M)$. Let p be a fixed point of the PARFUM algorithm with $p_{ij\dots n} > 0$ for all $i = 1, \dots, K; \dots, n = 1 \dots K$; then $p(\mathcal{A}_i^j) = q_j$, for $i = 1, \dots, M$ and $j = 1, \dots, K$.*

⁵This has been observed without exception on a great many numerical experiments, but not yet proved.

Proof: Let $p_{\dots,j,\dots}$ denote summation over all indices other than j , then we must show $p(\mathcal{A}_j^i) = p_{\dots,j,\dots} = q_j$. Since p is a fixed point we have (there are M terms in brackets on the left hand side):

$$Mp_{ij\dots n} = p_{ij\dots n} \left(\frac{q_i}{p_{i,\dots}} + \frac{q_j}{p_{\dots,j,\dots}} + \dots + \frac{q_n}{p_{\dots,n}} \right). \quad (4)$$

Since $p_{ij\dots n} > 0$, we may divide both sides by $p_{ij\dots n}$ and rearrange terms to obtain: $(M - \frac{q_j}{p_{\dots,j,\dots}} - \dots - \frac{q_n}{p_{\dots,n}})p_{i,\dots} = q_i$. Sum both sides over i and find $(M - \frac{q_j}{p_{\dots,j,\dots}} - \dots - \frac{q_n}{p_{\dots,n}}) = 1 = \frac{q_i}{p_{i,\dots}}$. The proof is concluded by noting that the above argument may be applied to any index \square .

It is easy to see that a fixed point of PARFUM can satisfy $p_{\dots,j,\dots} > 0$ only if this also holds for the starting point. PARFUM algorithm simply loops through the sample over and over again. The whole sample need not be loaded in memory, but can be loaded piece wise. Hence, there is effectively no restriction on sample size. We note that samples i and j which fall in the same interquantile intervals, for Y_1, \dots, Y_M , will be modified in exactly the same way. Hence, if we begin with the uniform vector $p_i = 1/N$, then the iterated values for these samples will always be equal.

Iterative proportional fitting The iterative proportional fitting (IPF) algorithm was introduced by [8] and [5]. The PARFUM algorithm adapts a starting measure to each partition, and then averages these adaptations to obtain the next iterate. The IPF, in contrast adapts a starting vector to the first partition, then adapts this to the second partition, and so on, cycling through the set of partitions. A complete step thus involves adapting the current measure to all M partitions. Hence if we have arrived at vector p , the next iteration is defined as

$$p' = (\dots ((p^{A^1})^{A^2} \dots)^{A^M}$$

This algorithm was first used to estimate cell probability in a contingency table subject to certain marginal constraints. Again, it is easy to see that if p satisfies constraints \mathcal{C} , then p is a fixed point of this algorithm.

The convergence of the IPF algorithm has been studied by many authors (for references, see Fienberg [7], and Csiszar [4]). It is not difficult to prove that if the IPF algorithm converges, then it converges to the I-projection of the starting probability vector on the set of probability vectors satisfying the constraints. Further [4] showed that starting with a probability vector p , IPF converges if and only if there is a vector r satisfying the constraints \mathcal{C} and satisfying:

$$p(\mathcal{A}_{i_1}^1 \cap \mathcal{A}_{i_2}^2 \dots \cap \mathcal{A}_{i_M}^M = 0) \iff r(\mathcal{A}_{i_1}^1 \cap \mathcal{A}_{i_2}^2 \dots \mathcal{A}_{i_M}^M = 0); \quad (5)$$

$$\mathcal{A}_{i_k}^j \in \mathcal{A}^j; k = 1, \dots, M. \quad (6)$$

4 Conclusion

Experience with the IPF algorithm shows that it converges much faster than the iterative PARFUM algorithm, *if* it converges. In some cases it will not converge, but oscillate. In applications to

probabilistic inversion using sample re-weighting, we frequently encounter infeasible problems, and in these cases IPF procedure doesn't converge. In practice a hybrid algorithm based pruning and switching to iterative PARFUM in case of oscillation might be attractive.

References

- [1] R.M. Cooke B. Kraan. Post processing techniques for the joint cec/usnrc uncertainty analysis of accident consequence codes. *J. Statist. Comput. Simul.*, 57:243–259, 1997.
- [2] R.M. Cooke B.C.P. Kraan. Processing expert judgements in accident consequence modeling. *Radiation Protection Dosimetry*, 90(3):311–315, 2000.
- [3] R.M. Cooke. Parameter fitting for uncertain models: modelling uncertainty in small models. *Reliability Engineering and System Safety*, 44:89–102, 1994.
- [4] I. Csiszar. I-divergence geometry of probability distributions and minimization problems. *Ann. of Probab.*, 3:146–158, 1975.
- [5] W.E. Deming and F.F. Stephan. On a last squares adjustment of sampled frequency table when the expected marginal totals are known. *Ann. Math. Statist.*, 11:427–444, 1940.
- [6] R.M. Cooke S. Hora M. Young J. Pasler-Ssauer L. Miller B. Kraan C. Lui M. McKay J. Helton A. Jones F. Harper, L. Goossens. Joint usnrc cec consequence uncertainty study: Summary of objectives, approach, application, and results for the dispersion and deposition uncertainty assessment. Technical Report VOL. III, NUREG/CR-6244, EUR 15755 EN, SAND94-1453, 1994.
- [7] S.E. Fienberg. An iterative procedure for estimation in contingency tables. *Ann. of Math. Statist.*, 41:907–917, 1970.
- [8] J. Kruithof. Telefoonverkeersrekening. *De Ingenieur*, 52(8):E15–E25, 1937.
- [9] S. Kullback. A lower bound for discrimination information in terms of variation. *IEEE Trans. Information Theory*, IT-13:126–127, 1967.
- [10] B.C.P. Kraan. R.M.Cooke. Uncertainty in compartmental models for hazardous materials - a case study. *J. of Hazardous Materials*, 71:253–268, 2000.