From Risk Analysis to Adversarial Risk Analysis

Part IV. Decision analytic computations

David Ríos

david.rios@icmat.es

AXA-ICMAT Chair in ARA and Royal Academy of Sciences

DRI. Aalto

Bayesian computational methods

In general, we shall need to compute (posterior) maximum expected utility alternatives

$$\max_a \int u(c(a,\theta))p(\theta|x)d\theta$$

Sometimes, it may be convenient to solve

$$\max_a \int u(a,\theta)p(x|\theta)p(\theta)d\theta$$

One possibility, approximate expected utilities by Monte Carlo then optimise the MC sums… Sampling from the posterior??

1. Select a sample $\theta^1, ..., \theta^m \sim p(\theta|x)$.

2. Solve the optimisation problem

$$\max_{a\in\mathcal{A}} \frac{1}{m}\sum_{i=1}^{m} u(a,\theta^i)$$

yielding $a_m(\theta)$.

Computational methods: Gibbs sampler

In some contexts, we are not able to sample directly from the target distribution, but we may sample from the marginal conditionals. Then, under appropriate conditions, the following scheme is designed to converge to the target distribution

1. Choose initial values $(\theta_2^0, \ldots, \theta_k^0)$. $i = 1$

2. Until convergence is detected, iterate through

   . Generate $\theta_1^i \sim \theta_1 | \theta_2^{i-1}, .., \theta_k^{i-1}$

   . Generate $\theta_2^i \sim \theta_2 | \theta_1^i, \theta_3^{i-1}, ..., \theta_k^{i-1}$

   . ...

   . Generate $\theta_k^i \sim \theta_k | \theta_1^i, ..., \theta_{k-1}^i$.

   . $i = i + 1$

Computational methods: Gibbs sampler

Imagine we need to sample from

$$p(\theta_1, \theta_2 \mid x) = \frac{1}{\pi} \exp\{-\theta_1(1 + \theta_2^2)\}$$

The conditionals are easily identified and a Gibbs sampler scheme is

1. Choose initial value for θ_2; e.g., the posterior mode, $\theta_2^0 = 0$. $i = 1$
2. Until convergence, iterate through
   - Generate $\theta_1^i = \mathcal{E}/(1 + [\theta_2^{i-1}]^2)$, ($\mathcal{E}$, standard exponential).
   - Generate $\theta_2^i = Z/\sqrt{2\theta_1^i}$, ($Z$, standard normal).

Computational methods: Metropolis sampler

Sometimes, we cannot sample from the conditionals. However, as we know, up to a constant, the target distribution, by choosing an appropriate candidate generating distribution q(.|.), under appropriate conditions, the following scheme is designed to converge to the target distribution

1. Choose initial values θ^0. $i = 0$
2. Until convergence is detected, iterate through
   - Generate a candidate $\theta^* \sim q(\theta|\theta^i)$.
   - If $p(\theta^i)q(\theta^i \mid \theta^*) > 0$, $\alpha(\theta^i, \theta^*) = \min\left(\frac{p(\theta^*)q(\theta^*|\theta^i)}{p(\theta^i)q(\theta^i|\theta^*)}, 1\right)$;
   - else, $\alpha(\theta^i, \theta^*) = 1$.
   - Do

$$\theta^{i+1} = \begin{cases} \theta^* & \text{with prob } \alpha(\theta^i, \theta^*), \\ \theta^i & \text{with prob } 1 - \alpha(\theta^i, \theta^*) \end{cases}$$

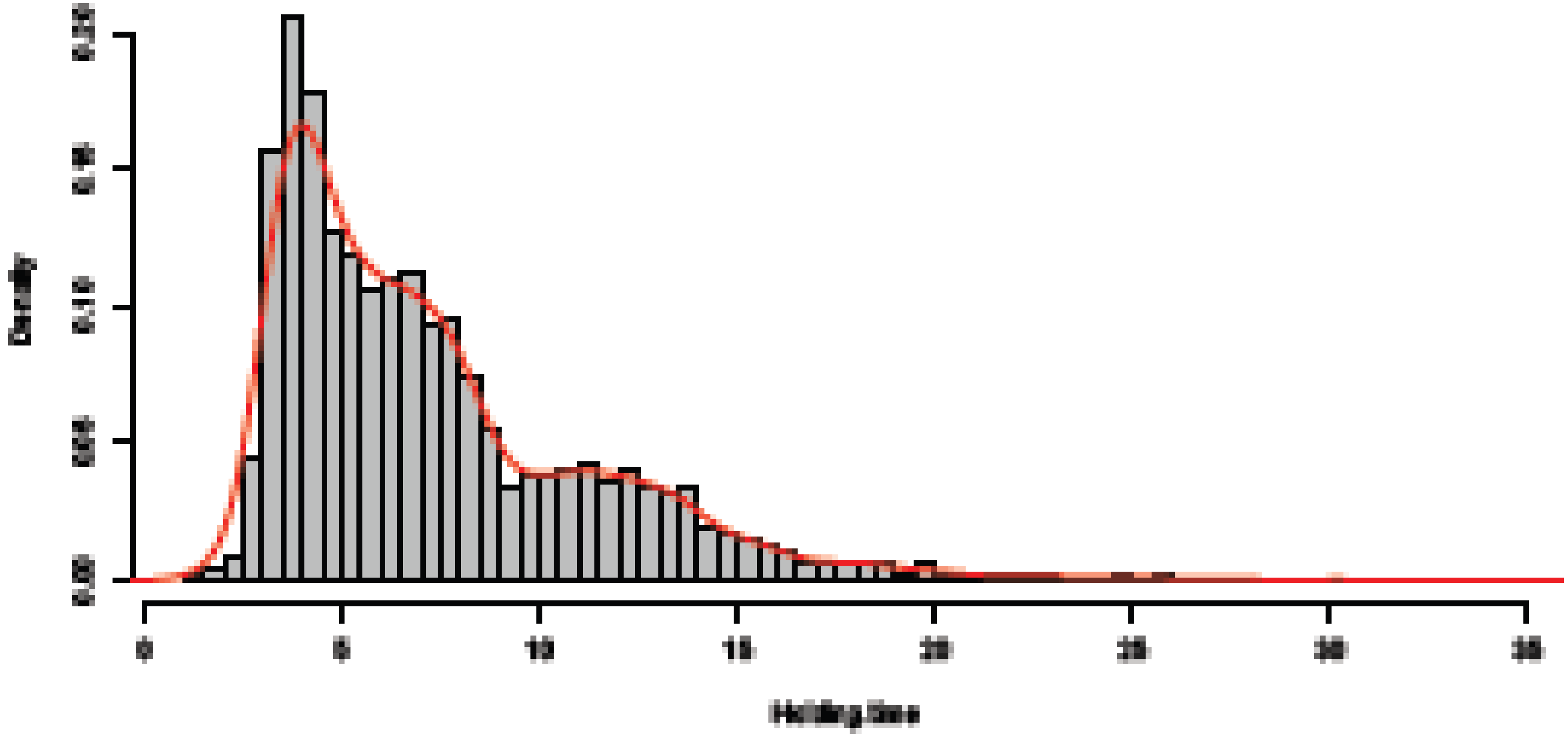
   - $i = i + 1$.

Mixtures of distributions

Modelling with mixtures provides a sound and flexible way to model uncertainty

- Theoretically. Any positive distribution may be approximated by a mixture of gamma distributions; any distribution may be approximated by a mixture of normal distributions➔An approach to density estimation.
- Computationally, there are ways to proceed via Markov chain Monte Carlo samplers (including uncertain number of components in mixture)
- In applications, describe model hetereogeneity (clustering), describe model uncertainty,…

Mixtures of distributions



Mixtures of distributions

Consider the mixture of exponentials (with Dirichlet-gamma priors)

$$f(t|\theta) = q_1\mu_1 \exp(-\mu_1 t) + \ldots + q_k\mu_k \exp(-\mu_k t)$$

By introducing labels describing the component mixture

$$\mathbf{z}_j|\mathbf{q},\boldsymbol{\mu} \sim \mathcal{M}_k(1; q_1, ..., q_k) \qquad t_j|\mathbf{z}_j,\mathbf{q},\boldsymbol{\mu} \sim \mathcal{E}(\prod_{i=1}^{k} \mu_i^{z_{ij}}).$$

We deduce the posterior conditionals, from which a Gibbs sampler follows

$$\mathbf{z}_j|t_j,\mathbf{q},\boldsymbol{\mu} \sim \mathcal{M}_k\left(1; \frac{q_1\mu_1\exp(-\mu_1 t_j)}{\sum_{i=1}^{k} q_i\mu_i\exp(-\mu_i t_j)}; ...; \frac{q_k\mu_k\exp(-\mu_k t_j)}{\sum_{i=1}^{k} q_i\mu_i\exp(-\mu_i t_j)}\right), j = 1, ..., n_s$$

$$\mu_j|\mathbf{t},\mathbf{z} \sim \mathcal{G}\left(a_j + \sum_{i=1}^{n} z_{ji}t_i, p_j + \sum_{i=1}^{n} z_{ji}\right), j = 1, ..., k$$

$$\mathbf{q}|\mathbf{t},\mathbf{z} \sim \mathcal{D}\left(\alpha_1 + \sum_{i=1}^{n} z_{1i}, ..., \alpha_k + \sum_{i=1}^{n} z_{ki}\right)$$

Mixtures of distributions

1. Start with arbitrary values $(\mathbf{q}^0, \mu^0, \mathbf{z}^0)$, $i = 0$.
2. Until convergence, iterate through
   - Generate $\mathbf{z}_j^{i+1} \sim \mathbf{z}_j | t_j, \mathbf{q}^i, \mu^i$, $j = 1, \ldots, n_s$.
   - Generate $\mathbf{q}^{i+1} \sim \mathbf{q} | \mathbf{t}, \mathbf{z}^{i+1}$.
   - Generate $\mu_j^{i+1} \sim \mu_j | \mathbf{t}, \mathbf{z}^{i+1}$, $j = 1, \ldots, k$.
   - Set $i = i + 1$.

Extended to an unknown number of components

Computational methods: Augmented probability simulation

Frequently, the involved posterior depends on decision made. The following observation helps in this context. Define an artificial distribution such that (u, needs to be nonnegative)

$$h(a,\theta) \propto u(a,\theta) \times p_\theta(\theta \mid x, a).$$

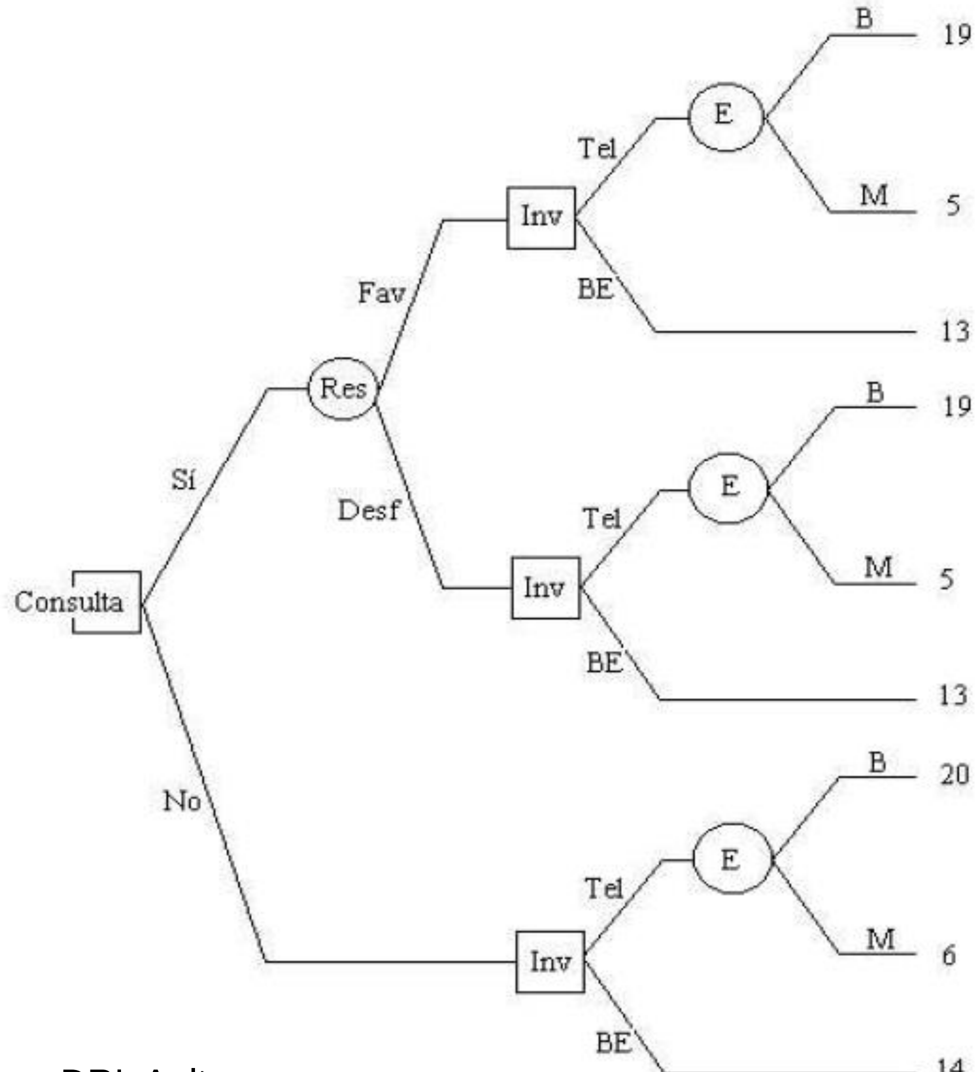
Then, the marginal of the artificial is proportional to expected utility

$$h(a) = \int h(a,\theta)\, d\theta_{a,\theta} \propto \Psi(a).$$

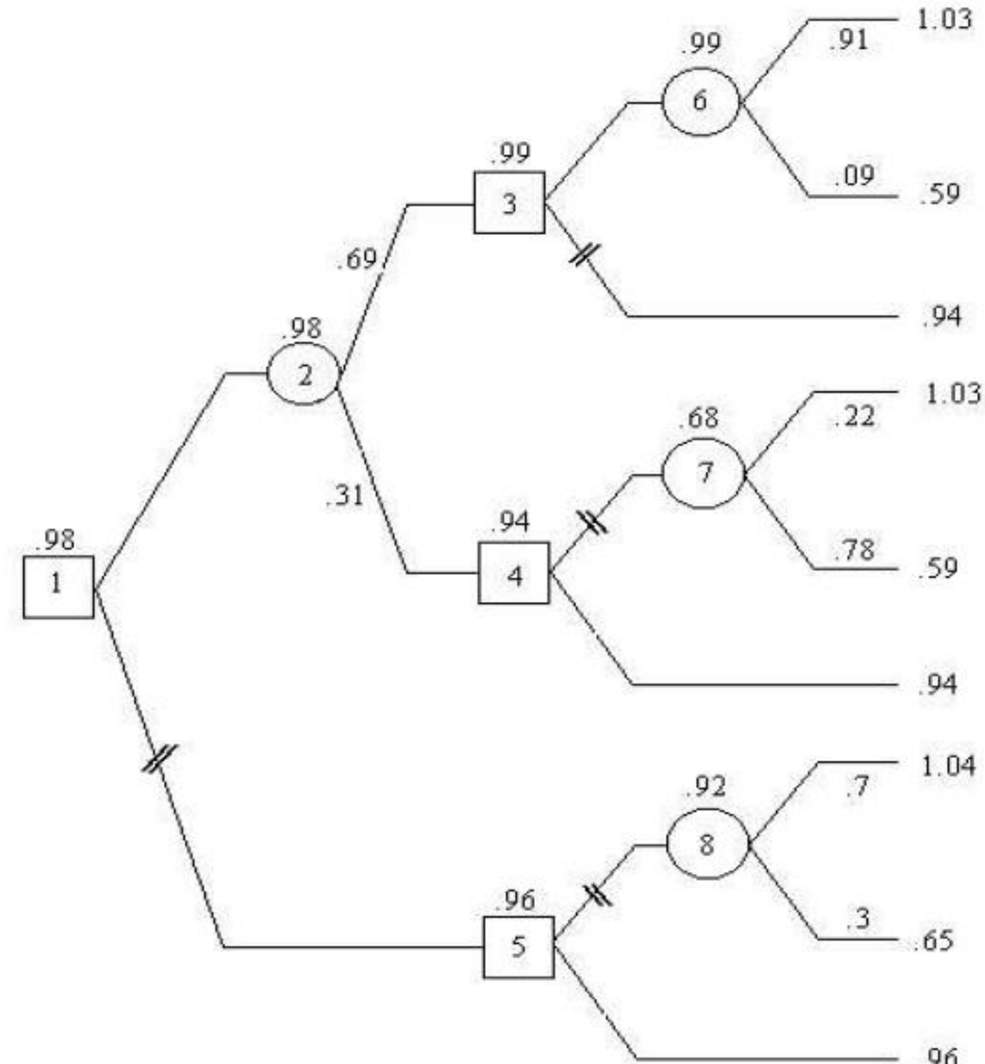
This suggests the scheme

1. Generate a sample $((\theta^1, a^1), ..., (\theta^m, a^m))$ from density $h(a,\theta)$.
2. Convert it to a sample $(a^1, ..., a^m)$ from the marginal $h(a)$.
3. Find the sample mode.

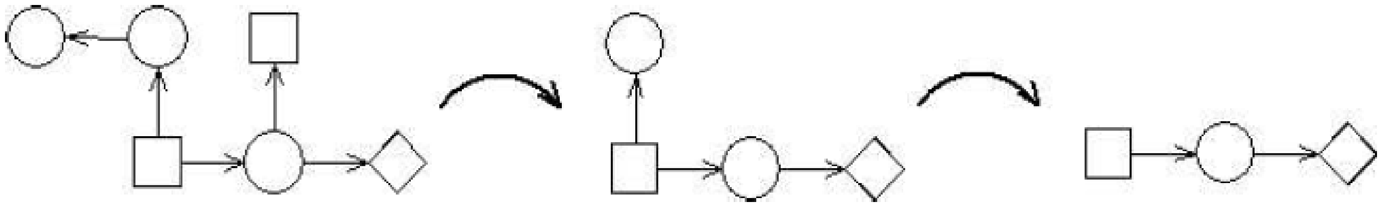
Dynamic programming



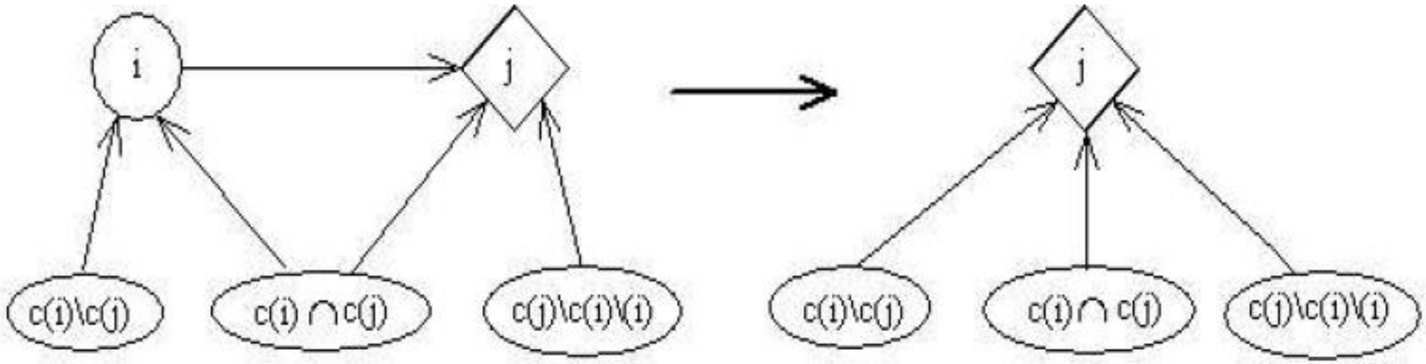
Dynamic programming

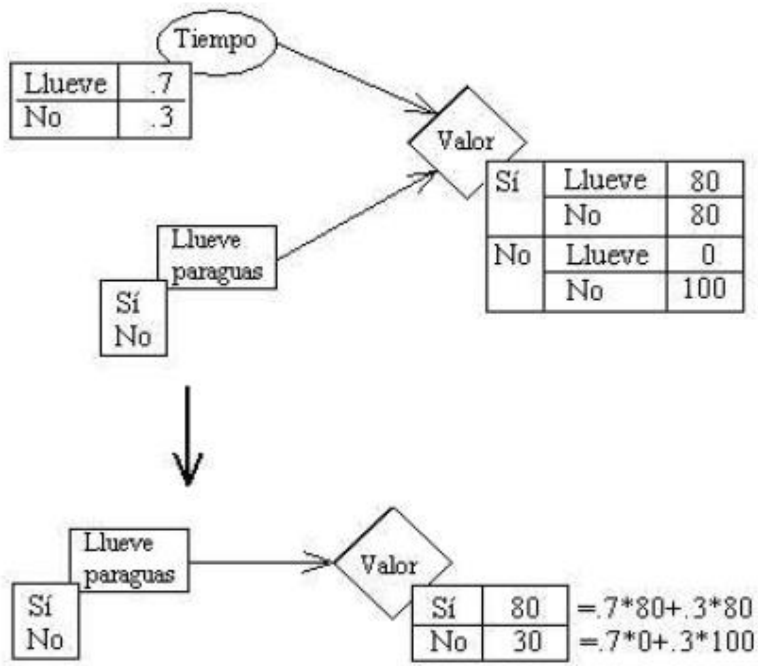


ID evaluation. Barren node elimination

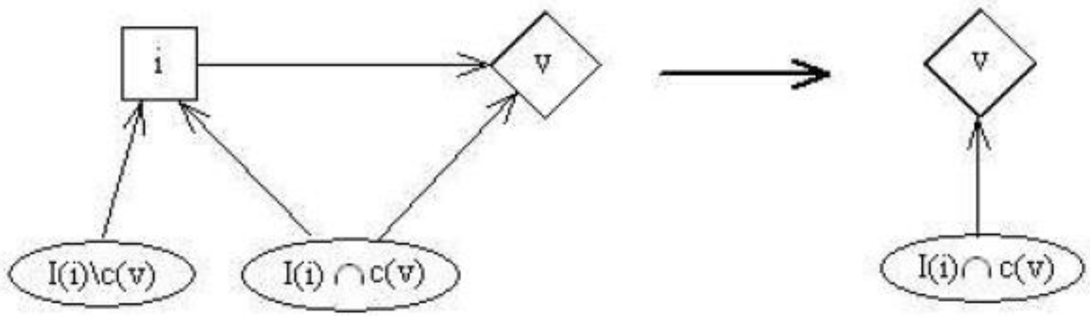


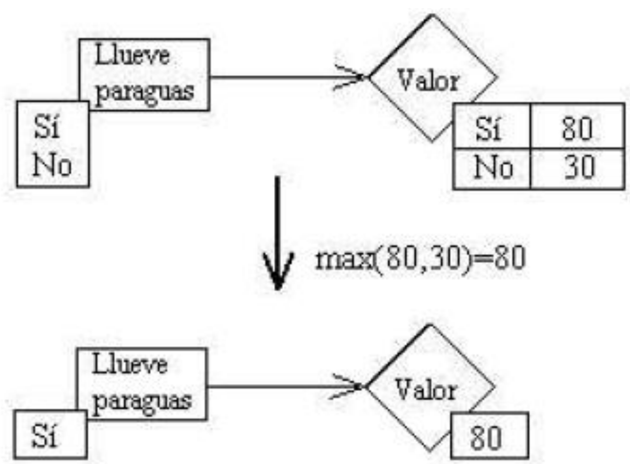
ID evaluation. Chance node elimination



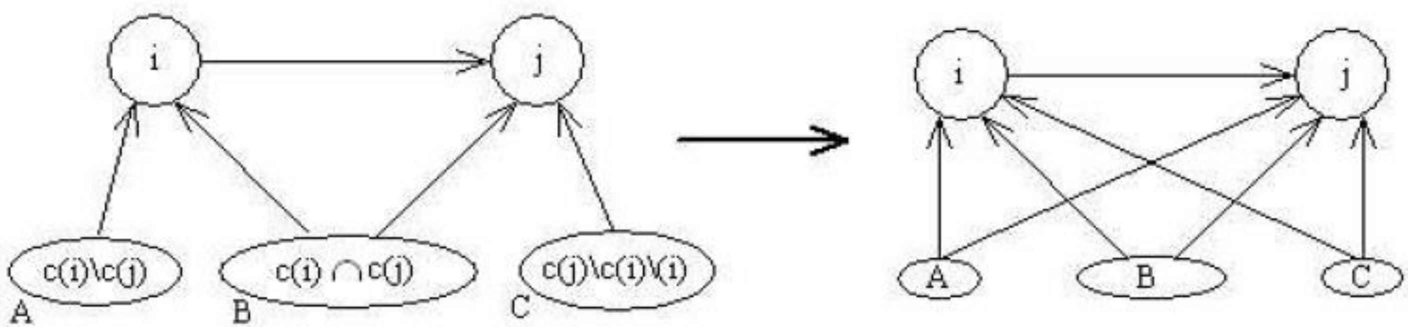


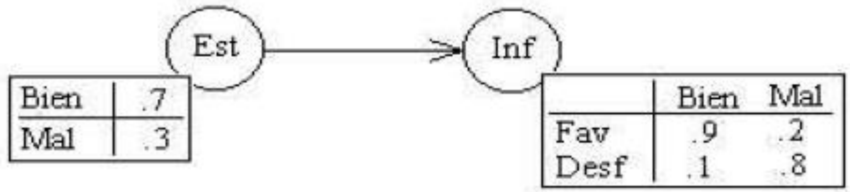
ID evaluation. Decision node elimination



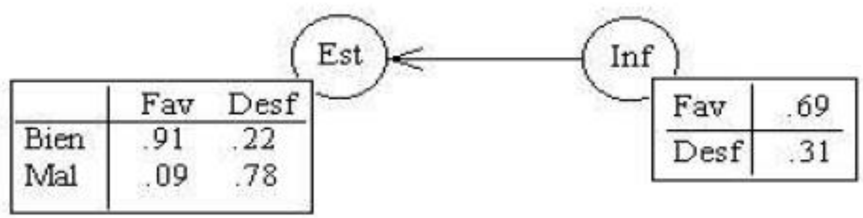


ID evaluation. Arc reversal





.r el arco



ID evaluation

For a well defined ID

- Eliminate barren nodes
- While value node has antecessors
  - If dec. node reducible, reduce it, eliminate barren nodes
  - Else if chance node reducible, reduce it
  - Else, find an invertible arc and invert it

ID evaluation.