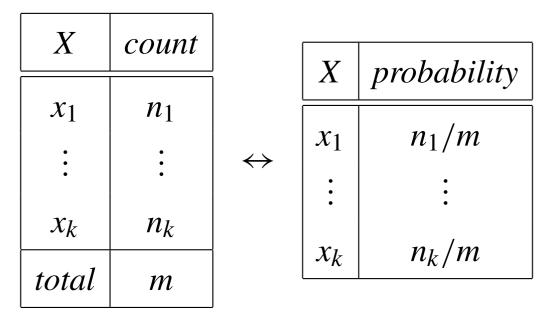
**Stochastic Simulation** 

 $\blacktriangleright$  Idea: probabilities  $\leftrightarrow$  samples

Get probabilities from samples:



If we could sample from a variables (posterior) probability, we could estimate its (posterior) probability.

## Generating samples from a distribution

- For a variable X with a discrete domain or a (one-dimensional) real domain:
  - $\succ$  Totally order the values of the domain of *X*.
  - Senerate the cumulative probability distribution:  $f(x) = P(X \le x).$
  - > Select a value y uniformly in the range [0, 1].
  - > Select the *x* such that f(x) = y.

## Forward sampling in a belief network

- Sample the variables one at a time; sample parents of *X* before you sample *X*.
- Given values for the parents of X, sample from the probability of X given its parents.

## **Rejection Sampling**

- To estimate a posterior probability given evidence  $Y_1 = v_1 \land \ldots \land Y_j = v_j$ :
- If, for any *i*, a sample assigns  $Y_i$  to any value other than  $v_i$  reject that sample.
- The non-rejected samples are distributed according to the posterior probability.

## **Importance Sampling**

- If we can compute P(evidence|sample) we can weight the (partial) sample by this value.
- To get the posterior probability, we do a weighted sum over the samples; weighting each sample by its probability.
- We don't need to sample all of the variables as long as we weight each sample appropriately.
- > We thus mix exact inference with sampling.



- Suppose the evidence is  $e_1 \wedge e_2$  $P(e_1 \wedge e_2 | sample) = P(e_1 | sample) P(e_2 | e_1 \wedge sample)$
- After computing  $P(e_1 | sample)$ , we may know the sample will have an extremely small probability.
- Idea: we use lots of samples: "particles". A particle is a sample on some of the variables.
- > Based on  $P(e_1 | sample)$ , we resample the set of particles. We select from the particles according to their weight.
- Some particles may be duplicated, some may be removed.