Table operations in the `gRain` package

Søren Højsgaard

February 28, 2008

Contents

1 Potentials and operations on these

In this section we describe the basic constructs for representing potentials and for making algebraic manipulations on these.

Consider a set $\Delta = \{\delta_1, \dots, \delta_R\}$ of discrete variables where δ_r has a finite set I_r of levels. Let $|I_r|$ denote the number of levels of δ_r and let $i_r \in I_r$ denote a value of δ_r. A configuration of the variables in Δ is then $i = (i_1, \dots, i_R) \in I_1 \times \ldots \times I_R$. The total number of configurations is then $|\Delta| = \prod_r |I_r|$. Let U be a non–empty subsets of Δ with configurations I_U and let i_U denote a specific configuration.

A potential ϕ_U defined on I_U is a non–negative function, i.e. $\phi_U(i_U) \geq 0$ for all $i_U \in I_U$. Let U and V be non–empty subsets of Δ with configurations I_U and I_V and let ϕ_U and ψ_V be corresponding potentials.

The product/quotient of ϕ_U and ψ_V is a potential defined on $U \cup V$ given by

$$\phi_{U \cup V} := \phi_U \times \psi_V \text{ and } \phi_{U \cup V} := \phi_U / \psi_V$$

with the convention that $0/0 = 0$. If $V \subset U$ is non–empty[1] then marginalization of ϕ_U onto V is defined as

$$\phi_V := \sum_{U \setminus V} \phi_U$$

[1]Marginalization onto an empty set is not implemented.

1.1 Implementation of potentials

Potentials are represented by `ctab` objects which are defined in `gRain`.

Given a set $U = \{v_1, \dots, v_S\}$, a potential ϕ_U is represented by i) the set U, ii) the levels $\{I_1, \dots, I_S\}$ and iii) a vector containing the values $\phi_U(u)$ with the convention that the first variable in U varies fastest. This effectively corresponds to a tree representation with the last variable v_S being the root of the tree.

1.2 Examples

`ctab` objects can be created as:

```
> yn <- c("y", "n")

[1] "y" "n"

> a.1 <- ctab("asia", list(yn), values = c(1, 99))

  asia potential
y    y         1
n    n        99

> t.a.1 <- ctab(c("tub", "asia"), list(yn, yn), values = c(5,
+     95, 1, 99))

  tub asia potential
1   y    y         5
2   n    y        95
3   y    n         1
4   n    n        99
```

Note that the first index varies fastest.

Tables can be normalized in two ways: Either the values are normalized over all configurations to sum to one as

```
> a.2 <- ctab("asia", list(yn), values = c(1, 99), normalize = "all")

  asia potential
y    y      0.01
n    n      0.99
```

Alternatively normalization can be over the first variable for *each* configuration of all other variables as

```
> t.a.2 <- ctab(c("tub", "asia"), list(yn, yn), values = c(5,
+     95, 1, 99), normalize = "first")

  tub asia potential
1   y    y      0.05
2   n    y      0.95
3   y    n      0.01
4   n    n      0.99
```

1.3 Operations on potentials

Multiplication and division of potentials is implented as follows. Consider multiplication of ϕ_U and ψ_V.

The vectors, say T_U and T_V, containing the values of the potentials are given a dimension attribute, i.e. are turned into arrays. Assume first that $V \subset U$. Then we reorder the elements of U to match with those of V, symbolically as $(V, U \setminus V)$ and permute T_U into $T_{V,U\setminus V}$ accordingly. This operation is fast with the `aperm()` function which is implemented in C. We can then form the product $T_{V,U\setminus V} T_V$ where T_V is recycled to match the length of $T_{V,U\setminus V}$. If V is not a subset of U then we expand the domain of ϕ_U into $\phi_{V,U\setminus V}$ and proceed as above. Marginalization is similarly based on permuting the array and then carrying out the relevant summations.

1.4 Examples

Hence we can calculate the joint, the marginal and the conditional distributions as

```
> ta.1 <- ctabmult(t.a.1, a.1)
> ta.1
```

```
  asia tub potential
1    y   y         5
2    n   y        99
3    y   n        95
4    n   n      9801
```

```
> ctabmarg(ta.1, "tub")
```

```
  tub potential
y   y       104
n   n      9896
```

```
> ctabdiv(ta.1, ctabmarg(ta.1, "tub"))
```

```
  tub asia   potential
1   y    y 0.048076923
2   n    y 0.009599838
3   y    n 0.951923077
4   n    n 0.990400162
```

The `ctab` function takes a `smooth` argument which by default is 0. A non-zero value of `smooth` implies that zeros in `values` are replaced by the value of smooth – before any normalization is made, e.g.

```
> ctab(c("tub", "asia"), list(c("y", "n"), c("y", "n")), values = c(0,
+     95, 0, 99), normalize = "first", smooth = 1)
```

```
  tub asia  potential
1   y    y 0.01041667
2   n    y 0.98958333
3   y    n 0.01000000
4   n    n 0.99000000
```