"Table": a tabulated prior

This prior allow the user to submit a prior for θ in a tabulated form, which is then interpolated to evaluate $\log \pi(\theta)$ as a continous function of the corresponding θ. Let

$$\theta_1, \theta_2, \ldots, \theta_m$$

be m values for θ with corresponding log-prior density

$$\log \pi(\theta_1), \log \pi(\theta_2), \ldots, \log \pi(\theta_m).$$

To define this as a prior in `R-INLA`, define *one* object of type `character`, with content

$$\text{table: } \theta_1\ \theta_2\ \ldots\ \theta_m\ \log \pi(\theta_1)\ \log \pi(\theta_2)\ \ldots\ \log \pi(\theta_m)$$

and use this as the name for the prior.

Example

This example define a loggamma-prior as the prior for the log-precision in three different ways.

```
rprior.func = function(lprec) {
    return (dgamma(exp(lprec), a, b, log = TRUE) + lprec)
}
rprior <- inla.rprior.define(rprior.func, a = 1, b = 0.1)

prior.expression = "expression:
            a = 1;
            b = 0.1;
            precision = exp(lprec);
            logdens = log(b^a) - lgamma(a)
                      + (a-1)*lprec - b*precision;
            ljacobian = lprec;
            return(logdens + ljacobian);"

prior.func = function(lprec) {
    a = 1; b = 0.1;
    return (dgamma(exp(lprec), a, b, log = TRUE) + lprec)
}
lprec = seq(-10, 10, len=1000)
prior.table = paste(c("table:", cbind(lprec, prior.func(lprec))),
        sep = "", collapse = " ")

n = 100
y = rnorm(n)

r = inla(y~1,
        data = data.frame(y),
        control.family = list(
                hyper = list(
                        prec = list(
                                prior = "loggamma",
                                param = c(1, 0.1)))))
```

```
rr = inla(y~1,
        data = data.frame(y),
        control.family = list(
                hyper = list(
                        prec = list(
                                prior = prior.expression))))

rrr = inla(y~1,
        data = data.frame(y),
        control.family = list(
                hyper = list(
                        prec = list(
                                prior = prior.table))))

rrrr = inla(y~1,
        data = data.frame(y),
        control.family = list(
                hyper = list(
                        prec = list(
                                prior = rprior))))

round(c(r$mlik[1], rr$mlik[1], rrr$mlik[1], rrrr$mlik[1]), 5)
```

Notes

- If the internal optimiser in `R-INLA` needs to evaluate the (log-)prior outside the domain given, it will stop and give an error.