# The Berkson Measurement Error (MEB) model

## Parametrization

This is an implementation of the Berksom measurement error model for a fixed effect. The observed covariate is $w$ but it is $x$ that goes into the linear predictor

$$\eta = \ldots + \beta x + \ldots \ ,$$

where $x = w + u$. The error term $u$ is Gaussian with prior $\mathcal{N}(0, \tau_u \mathbf{D})^1$, where $\tau_u$ is the observational precision of the error $\mathrm{Prec}(u|x))$ with possible heteroscedasticy, encoded in the entries $d_i$ of the diagonal matrix $\mathbf{D}$. The vector $s$ contains the fixed scalings $s = (d_1, \ldots, d_n)$ (with $n$ the number of data points).

## Hyperparameters

This model has 2 hyperparameters, $\theta = (\theta_1, \theta_2)$. The hyperparameter specification is as follows:

$$\theta_1 = \beta$$

and the prior is defined on $\theta_1$,

$$\theta_2 = \log(\tau_u)$$

and the prior is defined on $\theta_2$.

## Specification

The MEB is specified inside the `f()` function as

```
f(w, [<weights>,] model="meb", scale = <s>, values= <w>, hyper = <hyper>)
```

Here, `w` are the observed covariates, and the fixed scaling of the observational precision is given in argument `scale`. If the argument `scale` is not given, then $s$ is set to 1.

Note that only the unique values of `w` are used, so if two or more elements of `w` are *identical*, then they refer to the *same* element in the covariate $x$. If data points with identical $w$ values belong to different $x$ values (e.g., different individuals), please add a *tiny* random value to `w` to make this difference obvious to the model.

### Hyperparameter specification and default values

**doc** Berkson measurement error model

**hyper**

> **theta1**
>> **hyperid** 3001
>> **name** beta
>> **short.name** b
>> **prior** gaussian
>> **param** 1 0.001
>> **initial** 1
>> **fixed** FALSE
>> **to.theta** function(x) x

---

[1]Note: The second argument in $\mathcal{N}(,)$ is the precision not the variance.

**from.theta** `function(x) x`

**theta2**

    **hyperid** `3002`

    **name** `prec.u`

    **short.name** `prec`

    **prior** `loggamma`

    **param** `1 1e-04`

    **initial** `6.90775527898214`

    **fixed** `FALSE`

    **to.theta** `function(x) log(x)`

    **from.theta** `function(x) exp(x)`

**constr** `FALSE`

**nrow.ncol** `FALSE`

**augmented** `FALSE`

**aug.factor** `1`

**aug.constr**

**n.div.by**

**n.required** `FALSE`

**set.default.values** `FALSE`

**pdf** `meb`

## Example

```
n = 100
beta = 2
w = rnorm(n)
prec.u = 1
prec.y = 1
## heteroscedastic scaling
s = runif(n,min=0,max=1)
## true but unobserved covariate
x = w + rnorm(n, sd = 1/sqrt(s*prec.u))
y = 1 + beta*x + rnorm(n, sd = 1/sqrt(prec.y))

## prior parameters
prior.beta = c(0, 0.0001)
prior.prec.u = c(10, 9/prec.u)
prior.prec.y = c(10, 9/prec.y)


formula = y ~ f(w, model="meb", scale=s, values=w,
    hyper = list(
        beta = list(
            prior = "gaussian",
```

```
            param = prior.beta,
            fixed = FALSE
        ),
        prec.u = list(
            prior = "loggamma",
            param = prior.prec.u,
            initial = log(prec.u),
            fixed = FALSE
        )
    )
)

r = inla(formula, data = data.frame(y, w, s),
    family = "gaussian",
    control.family = list(
        hyper = list(
            prec = list(param = prior.prec.y,
                fixed = FALSE
            )
        )
    )
)
```

## Notes

- `INLA` provides the posteriors of $\nu_i = \beta x_i$ and NOT $x_i$.

- The posteriors of $\nu_i$ come (default) in the order given by the sorted (from low to high) values of `w`. The entry `$ID` gives the mapping.

- The option `scale` defines the scaling in the same order as argument `values`. It is therefore adviced to also give argument `values` when `scale` is used to be sure that they are consistent.