

A couple of colleagues have alerted us to some issues with the [BCEA](#) function `evppi`, under MS Windows. Basically, it seemed as if under MS Windows and with R 4.0.0 (which is a recent, though not the most recent release of R) the `evppi` function would break, when trying to make the analysis based on [INLA](#) (and the work discussed [here](#)).

In particular, if the user didn't have the R package `INLA` already installed, when trying to run

```
x=evppi(..., method="INLA")
```

`BCEA` would oblige and “suggest” the installation of `INLA`. Now this is also a bit complicated, because `INLA` is a “heavy” package and, in addition, is not installable from the main [CRAN](#). Under Linux and (I think!) Mac OS, this isn't a problem and `BCEA` would simply tell the user to install `INLA` — though the link to the URL from which this can be done was outdated in the current `CRAN` release of `BCEA`. Under MS Windows, however, I think that things become a bit more complicated as, in order to install a package not on `CRAN`, another package is needed (`Rtools`). But `Rtools` in its most recent version (which would be aligned with R) wouldn't install directly. This then would make it impossible to install `INLA` and its main dependencies and therefore would break `evppi`.

I did a bit of digging and I found a solution (which my colleagues have confirmed has worked for them!). So, if:

- you are under MS Windows and have a version of R that is at least as recent as 4.0.0;
  - you have never installed `INLA`;
  - you have installed `BCEA` and now want to use `evppi` to compute the EVPPI for your model, using the fast algorithm based on `INLA`
- here's what you need to do:

1. Follow the instructions [here](#) to install the most recent version of `Rtools`;
2. Install `INLA` using the following code

```
install.packages("INLA", repos=c(getOption("repos"), INLA="https://inla.r-inla-download.org/R/stable"), dep=TRUE)
```

3. If you don't already have it installed, install the package `ldr` (which is needed to complement the fast EVPPI algorithm), using the code

```
install.packages('ldr')
```

and then you're good to go!