

Comments/Change for *tensorregress*

- **Package**

1. Installed *tensorregress_1.0.tar.gz*, the package can not use other packages. I add "import" terms in the NAMESPACE to enable our package to connect with other packages.
2. To let who use our package but not has installed related packages, I change first few lines in *brick.R* like this:

```
#library(rTensor)

if(!require(rTensor)){ install.packages("rTensor") library(rTensor) }
```

- **sim_data**

1. I add few warnings for `sim_data` when the inputs meet follow condition: (take mode 1 as example)

- i. `r1 < 0`

warning: *rank of core tensor should larger than 0*

- ii. `p1 > d1`

warning: *data can be generated, but to use the model, col dim of X_{cover} should NOT larger than $whole_shape$*

- iii. `block[1] = F & p1 < r1`

warning: *col dim of the X_{cover1} should LARGER than rank 1 of the core tensor*

- iv. `block[1] = T & p1 < r1`

warning: *rank 1 of core tensor would degenerate to be equal or smaller than col dim the X_{cover1}*

- v. `block[1] = T & (p1 = 1 | r1 = 1)`

warning: *group of membership should larger than 1*

- vi. `b1=sort(sample(1:r1,p1,replace=TRUE)); length(unique(b1)) == 1`

warning: *rank 1 degenerate to 1*

2. The output of `sim_data`, each tensor in `tsr` and `c_ts` is `array`, while `G,U` are `tensor`. Should we uniform the class of the outputs?

- **tensor_regress**

1. Due to adding constrain as `vanilla_penalty` would lead the log-likelihood decrease, the algorithm may stop just in one iteration without convergence. Should we deliberately keep it run more iterations in this case?

- **sele_rank**

1. `sele_rank` allows adding constrains, however, it just use the default parameters of the constrains (for example `lambda = 0.1, alpha = 1`). Should we add inputs of the function to adjust the parameter?