**Navodila za uporabo R interface in testiranje**

V repozitoriju sem vse stare verzije shranil v podmape »archive«. V mapi R so sedaj 3 funkcije:

* df\_to\_list\_new. Pomožna funkcija ki spremeni podatke iz data.frame v list, ustrezen za input v Stan.
* compare\_samples. Pomožna funkcija za testiranje, ki primerja porazdelitev dveh matrik vzorcev.
* sample\_from\_exe\_new. Glavna funkcija, predstavlja R interface.

**sample\_from\_exe\_new**

Funkcijo sem dokumentiral, tako da je najbolje pogledati funkcijo za podrobnosti in nek primer iz testiranja za uporabo. Na kratko:

* V parameter df vstavimo data.frame s podatki.
* Izberemo ime modela (kako je shranjen exe, brez .exe končnice). Default je sedaj nastavljen na »new\_model«.
* Potem nastavimo imena stolpcev, ki vsebujejo ID pacienta, ID študije, čas, score in ali je študija placebo. Ta imena vstavimo brez narekovajev!
* Vstavimo kovariate za rate in baseline v obliki formule (~ A + B + ...), kjer so A, B,... imena stolpcev, ki vsebujejo kovariate.
* Določimo ali so kovariati aditivni ali multiplikativni z m\_r in m\_b.
* Če imamo model z 2 score-oma, potem naredimo enako še za drugi score.
* Izberemo gpu\_enabled (Davor?). Default je 1.
* Izberemo število vzorcev z num\_samples in num\_warmup.
* Poljubno spremenimo parametre vzorčevalnika.

Funkcija vrača seznam (list) dolžine 3:

1) R objekt stanfit.

2) Podatki, ki so bili uporabljeni za učenje modela (namenjeno testiranju in preverjanju).

3) Mape, ki mapirajo originalne ID pacientov in študij v ID-je uporabljene v modelu (glej Slack pogovor za podrobnosti, skratka preindeksirat je treba paciente in študije, če niso zaporedna cela števila).

**Testiranje**

Skripte za testiranje so v ./tests/generalized\_logistic\_model/. Imamo skripto test\_R\_interface, ki je namenjena samo testiranju tega, ali prenese pravilne podatke v model. To ne služi za primerjavo Rstan in cmdstan.

Ostale skritpe so za primerjavo Rstan in cmdstan, ali dveh cmdstan modelov. Vsaka ima nek specifičen input, opisano je na začetku skripte. Na začetku skripte vstaviš število testa (testn), zato da se vzorci shranjujejo glede na test. Na začetku zavihka »data input« se nastavi število vzorcev (warmup + sampling) in izbere podatke. Zaenkrat sem dodal še podatke o tem ali je študija placebo ročno, predvidevam da bo to enkrat v data.frame-u. V zavihku »compile model« izbereš ime modela, ki mora biti shranjen v mapi »Stan«. Potem enostavno poženeš skripto do zavihka »Parameter distributions«. To nauči oba modela in vzorce tudi shrani v mapo »saved\_samples«, da naslednjič ni več potrebno vzorčiti. Če želimo ponovno zagnati ta model, potem je potrebno ročno izbrisati .rds datoteko v tej mapi.

Zadnji trije zavihki so namenjeni primerjavi. Najprej so porazdelitve. V funkciji compare\_samples lahko nastaviš ali želimo povprečje (default) ali mediano vzorcev. Potem primerja ali je razlika med povprečjema znotraj dveh MCMC SE. Drugo so ESS. Tretje je Rhat. Tukaj je treba bit pozorn, ker če obstajajo Rhat ki so večji od 1 nakazuje da model ni konvergiral. Mogoče je najbolje da se glede tega posvetujemo z Erikom, kakšne vrednosti so sprejemljive. Če ni konvergiral nima smisla primerjati porazdelitev.

Testiranje povzeto:

* Nastavi številko testiranja v zavihku na začetku skripte.
* Nastavi število vzorcev v zavihku »data input«.
* Nastavi podatke v zavihku »data input«.
* Nastavi ime modela v zavihku »compile model«.
* Poženi do zavihka »Parameter distributions«.
* Preveri porazdelitve, ESS in Rhat.

Seveda se lahko te funkcije spremeni za primerjavo različnih modelov. Pri tem je potrebno paziti, da je input pri Rstan isti kot pri cmdstan.

Pri »Parameter distributions« vrne funkcija data.frame. Stolpec »is\_same« nakazuje ali so vrednosti znotraj 2 \* MCMCSE. Stolpec »diff\_at\_three« nakazuje, ali je razlika manjša kot na drugi decimalki. Lahko poljubno dodate kako drugo mero v funkcijo **compare\_samples**.