

Code Review Power Simulation

in zeile 32-34 hat die rep funktion keinerlei Auswirkung

zeilen 74 - 84 sind ein bisschen schwierig nachzuvollziehen, was da genau passiert. vermutlich gäbe es da noch eine elegantere möglichkeit (z.B. in dplyr oder data.table gruppieren nach subject & experiment und dann die neuen advice werte dort einfügen), aber ich traue mich nicht das spontan umzuschreiben ohne einen fehler zu machen und solange es stimmt ist es ja am ende egal. habs nochmal nachvollzogen und wenn ich es richtig verstehe, wird weiter oben erstmal nur unterschieden zwischen advice = 0 und advice = 1. in dem komplizierten block, werden dann innerhalb der personen und experimente alle 1er komplett ersetzt durch entweder 1 oder 2. vielleicht wäre es leichter zu lesen, wenn oben statt advice = 1 ein platzhalter eingeführt wird, z.B. advice = 99. dann wäre es vielleicht klarer, dass diese 99er unten nochmal vollständig ersetzt werden. dann könnte man auch einfach checken, dass wirklich alle 99er im datensatz danach weg sind.

die annahmen in zeile 109 - 126 wären eventuell noch leichter lesbarer, wenn man von oben her variablen definiert, diese kurz erklärt (z.B. `base_nonExpert_correctAdvice <- qlogis(0.75) #baseline prob for non experts with correct AI advice is 75%`) und diese Variablen dann immer wieder verwendet. aktuell ist es relativ schwer zu erkennen, welche teile der gleichungen immer jeweils von weiter oben kopiert wurden.

die double checks in zeile 129 - 162 funktionieren ja so, dass man das was da steht noch in plogis reinsteckt. das wird noch nicht so richtig klar. (noch weniger fehleranfällig bzgl. copy-paste errors wäre es natürlich, wenn diese checks die oben von mir vorgeschlagenen variablen verwenden würden.)

typo in zeile 169 (e.g., due due different difficulty of head scans)

ich nehme an die `nsim <- 20` sind aktuell nur um zu testen oder? ich wurde sagen für verlässliche aussagen bezgl. der zu erwarteten power muss das deutlich höher sein, vom gefühl her würde ich sagen mindestens sowas wie 200.

ab zeile 179 benötigt man das lme4 paket, das wurde aber nirgends geladen

habt ihr nachgelesen, ob diese normalverteilungs-CIs in zeile 225 - 226 (`est + 1.96 * stderr`) für die art von modelle zumindest „okish“ sind? ich kenne das eher so, dass die CIs bei GLMMs gebootstrapped werden, weil die einfachen keine guten eigenschaften haben.

eine sache ist mir noch aufgefallen: in zeile 48 werden ja die personen zufällig den bedingungen zugeordnet, aber natürlich bekommt man dadurch nicht exakt die gleiche anzahl pro gruppe. diese art von randomness wird aber bei der

simulation der AV später nicht berücksichtigt, da ja für jede simulation der y-variable immer der gleiche datensatz für die UV werte verwendet wird. das heißt streng genommen gilt dann auch das ergebnis für die power nur dann, wenn du genau die hier verwendete aufteilung der personen zu den gruppen hast. ich würde allerdings vermuten das ist hier good enough (weil es vermutlich keinen großen unterschied machen wird) und es ist nicht unbedingt notwendig, die zuteilung jedes mal neu zufällig zu ziehen, aber streng genommen wäre es notwendig, wenn beim tatsächlichen datensatz dann auch eine zufällige zuteilung vorgenommen wird, also die exakten gruppengrößen nicht vorher schon feststehen.