

Posudek na diplomovou práci **Žanety Vančurové**

Zpracování kompozičních dat v Matlabu

Diplomová práce se zabývá vytvořením programových kódů v Matlabu pro práci s kompozičními daty. Práce je rozčleněna do čtyř kapitol (nepočítaje úvod a závěr) a má celkem 72 stran. K práci je přiloženo CD s vytvořenými kódy v Matlabu.

V úvodu autorka uvádí cíl práce a seznamuje čtenáře s obsahem práce. První kapitola je věnována základům z teorie kompozičních dat a Aitchisonově geometrii na simplexu. Ve druhé kapitole jsou popsány tři základní transformace kompozičních dat. Ve třetí kapitole se autorka charakteristikami kompozičních dat, jejich centrováním a škálováním a konstrukcí biplotu. Všechny předchozí kapitoly obsahují M-soubory kódující popisované operace. Stručné shrnutí vytvořených kódů a porovnání s obdobnými kódy dostupnými ve statistickém softwaru R se nachází v poslední kapitole. V závěru autorka hodnotí naplnění cílů práce.

K práci mám tyto dotazy a připomínky (připomínky zpravidla začínají číslem stránky, horní resp. dolní index u čísla stránky pak značí, kolikátého řádku shora resp. zdola se připomínka či dotaz týká):

- 10₂ - Co se rozumí změnou měřítka?
- 11 - V definicích 1.2 a 1.3 není uvedeno, jakých hodnot může nabývat κ . Jelikož podoba simplexu a uzávěru vektoru závisí na hodnotě κ , bylo by dobré, aby se κ objevilo i v jejich označení (např. S_κ^D a C_κ) a stručnější označení S^D a C aby zůstalo pouze pro $\kappa = 1$. Nicméně to není chyba autorky, označení se takto běžně používá.
- 12⁴ - Podmínka `if (x<=0)` nebude vyhodnocena, tak jak autorka očekává, chyba bude vypsána jen v případě, že budou všechny složky $\mathbf{x}$ nekladné. Stejná chyba se vyskytuje i v šesti dalších funkcích, konkrétně: `perturb` (str. 15), `mocn_trans` (str. 17), `clr_trans` (str. 29), `ilr_trans` (str. 35), `alr_trans` (str. 42) a `alr_index` (str. 45).
- 15⁴ - Kromě toho, že jsou špatně zapsané podmínky pro test nekladných prvků (viz předchozí připomínka), tak v tomto případě se navíc test `if (y<=0)` provede jen pokud bude splněna podmínka `if (x<=0)`, což není správně.
- 19 - Pokud neřeknete, že $\kappa = D$ není zřejmé, proč by mělo platit $C(1, \dots, 1) = (\frac{1}{D}, \dots, \frac{1}{D})$.
- 20⁹⁻¹⁸ - Testování nezápornosti `if (x(a)<=0)` a `if (y(b)<=0)` se zbytečně provádí ve dvou do sebe vnořených cyklech, i když každá z podmínek závisí pouze na jedné z řídicích proměnných. Prázdnou větev `else` není třeba zapisovat. Podobná chyba je i ve funkci `vzdálenost` na str. 22.

- 20^{19–21} - Je zbytečné testovat dvě podmínky $Dx < Dy$ a $Dx > Dy$, když by stačila jedna $Dx \neq Dy$. Podobná zbytečná komplikace je i ve funkci `vzdálenost` na str. 22.
- 35¹ - Vůbec není zřejmé, jak vypadají vektory $\mathbf{v}_i$, kolik je v nich nenulových složek a jaké mají hodnoty.
- 35⁸ - Jak lze skládáním řádkových vektorů $\mathbf{v}_i$ za sebe získat matici $\mathbf{V}$?
- 37 - Zbytečná proměnná `puvodni`, když stačí použít jen proměnnou `z`.
- 38 - V popisu metody jsou použity proměnné y_i , v příkladu počítáte $\tilde{y}_i$, ale poté prvky x_i počítáte z proměnných y_i , které v příkladu nejsou.
- 38₄ - Jaký je význam věty „Postupné binární rozdělení je hierarchie složek kompozice“?
- 38–40 - Z popisu procesu binárního rozdělení není ani zmínka o tom, jak dělíme složky do skupin +1 a -1. Navíc později bez vysvětlení významu zavádíte i 0, byť mluvíte jen o dvou skupinách. V příkladu 2.4 se nedržíte značení zavedeného v popisu procesu.
- 42–43 - Funkce `alr_trans` a `alr_inv` jsou pouze speciální případy funkcí `alr_index` a `alrinv_index`. Pokud už tyto funkce chcete vytvářet, bylo by vhodnější v nich pouze zavolat obecnější funkce `alr_index` a `alrinv_index` s indexem nastaveným na poslední hodnotu.
- 43₈ - Příkazem `y(:,n+1)=1` zbytečně přidáváte k matici `y` dva sloupce. Pro původní hodnotu `n` to jen `(n+1)`. sloupec tvořený nulami a `(n+2)`. sloupec tvořený jedničkami. Přičemž druhý z nich vůbec k ničemu nepoužijete.
- 48₃ - Když uvádíte, že výhodou Alr transformace je snadná interpretovatelnost, bylo by dobré tuto interpretaci uvést.
- 51 - Domnívám se, že v příkladu 3.1 by bylo vhodnější vypočítat centrum pro stejnou hodnotu κ jakou mají data, neboť by pak byl názornější vztah této charakteristiky k datům.
- 54 – 55 - Je zbytečné psát kompletní kód funkce `T2`, stačilo by v ní využít volání funkce `T` a výsledek vydělit 2, podle vztahu na straně 55 dole.
- 56 – 57 - Ve funkci `tot_var` se zcela zbytečně počítá znovu variační matice, přičemž by stačilo zavolat funkci `T` a poté použít příkazy z posledních dvou řádků.
- 58 - Bylo by dobré uvést, k čemu jsou operace centrování a škálování užitečné.
- 59 - Domnívám se, že v příkladu 3.5 by bylo vhodnější vypočítat centrovanou matici pro stejnou hodnotu κ jakou mají data, neboť by pak byl názornější vztah původním k datům.

- 60¹⁰ - Z popisu není zřejmé, jak umocňujeme datový soubor **X**, zda jako celek tj. maticově, nebo zda umocňujeme jednotlivé prvky zvlášť.
- 60 – 61 - Velká část funkce `skala` počítá celkový rozptyl, pro který už existuje funkce `tot_var`. Využitím volání této funkce by bylo možné kód významně zkrátit.
- 63¹³ - Tvrzení, že nejlepší aproximací matice **Z** je matice **Z** mi nepřijde příliš objevené.
- 63 - Celý popis toho, jak získáme singulárního rozkladu mi přijde zmatený, obzvlášť když jej nakonec vypočítáte funkcí `svd` z Matlabu.
- 64 - Jsou i jiné možnosti, jak získat matice **G** a **H**? Z jakých důvodů se volí zrovna tento způsob?
- 66 – 67 - Na centrování dat a výpočet CLR transformace už máte hotové funkce, vhodnější by bylo je zde zavolat, místo opakování kódů.
- 65 - Co je to „ukazatel na pozici pro pozorování“? Obecný popis interpretace kompozičního biplotu je dost nejasný, u příkladu 3.7 na straně 68, který by mohl pomoci to objasnit, tento popis zcela chybí.
- 70 - Pokud mají napsané funkce v Matlabu sloužit jako ekvivalent funkcí v R, bylo by dobré se více snažit o zachování stejných nebo podobných jmen. Samozřejmě jen tam, kde to dovolí pravidla pro jména funkcí v Matlabu, a kde nedochází ke kolizi se jmény jiných funkcí Matlabu

Hlavním cílem práce bylo vytvořit v Matlabu funkce pro práci s kompozičními daty. To autorka sice splnila a rozhodně ne ideálně. Největší výtku bych měl k tomu, že některé části kódu se zbytečně opakují v řadě funkcí a autorka se vůbec nepokusila tyto společné části kódu osamostatnit do pomocných funkcí, které by z ostatních funkcí pouze volala. Kromě toho, co jsem uvedl už v předchozích připomínkách, bych navíc zmínil například opakující se kontroly nezápornosti dat a kontroly stejných řádkových součtů ve většině funkcí nebo velké shodné části funkcí `centrovani` a `centrum`. Navíc programové kódy obsahují i několik chyb týkající se kontroly správných hodnot vstupních dat. Další nedostatky (zejména nepřiliš jasné popisy některých postupů) se najdou i v textu práce. Pro praktické nasazení by také kód vyžadoval určitou optimalizaci s využitím zabudovaných funkcí a operátorů Matlabu místo množství cyklů, což by nejen zkrátilo kód, ale pro větší datové soubory by to i výrazně urychlilo výpočet. Práci doporučuji k obhajobě a navrhuji hodnocení klasifikačním stupněm **E**.

V Olomouci dne 19. 5. 2016.

Pavel Ženčák
oponent práce