

Kapitał społeczny - poprawione skale

Szymon Talaga

16.12.2014

Raport przedstawia drugie podejście do skalowania kapitału społecznego. Zmiana strategii skalowania została wymuszona tym, że otrzymane skale, choć dobrej jakości pod względem zgodności z modelem mokkena, były obciążone bardzo poważną skazą - były zbyt łatwe i nie dyskryminowały między osobami o wysokim kapitale. Dlatego zastosowano trochę inne podejście, ale oparte na wynikach pierwszej procedury skalowania, które ma temu zaradzić.

Jako punkt wyjścia przyjęto dokonane w poprzedniej analizie spostrzeżenie, że baterie pytań scnow i schelp odpowiadają innym wymiarom kapitału społecznego. Dlatego tym razem zostały wyodrębnione najlepsze skale w obrębie obu baterii, w których skład wchodziły tylko pytania o wyższym poziomie trudności. Dokładniej rzecz biorąc uwzględnione jedynie pozycje o średnich mniejszych bądź równych 3. Dzięki temu możliwe było wyłonienie skal o większej mocy dyskryminacyjnej.

Wczytanie niezbędnych pakietów, funkcji i danych:

```
library(mokken)
library(lattice)
library(knitr)
source("../Scaling/scaling-Helper.R")
source("../HelperFunctionsMisc/ComputingMisc.R")
load("../Scaling/Scal-ResGenAll.RData") # Wykorzystana zostaje przestrzeń zmiennych z poprzedniej anali.
load("../MainData/MainData10.RData")
D10 = D
```

Skalowanie mobilizacji zasobów (resmob)

Wybór i ograniczenie zbioru zmiennych tylko do tych, których średnie są mniejsze bądź równe 3:

```
D.help = D.back[, grep('schelp', names(D.back))]
diffs.help = apply(D.help, 2, mean)
D.hdiff = D.help[, names(diffs.help)[diffs.help <= 3]]
```

Skalowanie

Automatyczny dobór pozycji o wystarczająco dobrych współczynnikach skalowania w oparciu algorytm genetyczny:

```
scale.ga.hdiff = aisp(D.hdiff, search="ga", lowerbound=.35)
kable(scale.ga.hdiff)
```

	Scale
schelp4	1
schelp6	1
schelp7	0
schelp10	1
schelp14	1
1	

Ograniczenie zbioru zmiennych do najlepszej skali (skala 1):

```
D.hdiff = D.hdiff[, rownames(scale.ga.hdiff)[scale.ga.hdiff[,1] == 1]]
```

Jak widać pytania które pozostały odnoszą się do pomocy w zakresie: porady prawnej (4); załatwienia pracy dorywczej (6); napisania dobrych referencji (10); pożyczki większej sumy pieniędzy (14). Ten zestaw pytań jest o tyle dobry, że faktycznie odnosi się tylko i wyłącznie do dosyć specyficznych, i bardzo przydatnych, form pomocy, dzięki czemu, ma szanse faktycznie różnicować osoby badane.

Homogeniczna monotoniczność:

```
mono.hdiff = check.monotonicity(D.hdiff)
kable(summary(mono.hdiff))
```

	ItemH	#ac	#vi	#vi/#ac	maxvi	sum	sum/#ac	zmax	#zsig	crit
schelp4	0.40	4	0	0	0	0	0	0	0	0
schelp6	0.43	4	0	0	0	0	0	0	0	0
schelp10	0.45	4	0	0	0	0	0	0	0	0
schelp14	0.38	4	0	0	0	0	0	0	0	0

Widać, że dane dobrze odpowiadają założeniu homogenicznej monotoniczności.

Podwójna monotoniczność:

```
iio.hdiff = check.iio(D.hdiff) # Metoda niezmiennej kolejności pozycji
kable(summary(iio.hdiff)$item.summary)
```

	ItemH	#ac	#vi	#vi/#ac	maxvi	sum	sum/#ac	tmax	#tsig	crit
schelp14	0.38	3	0	0	0	0	0	0	0	0
schelp6	0.43	3	0	0	0	0	0	0	0	0
schelp4	0.40	3	0	0	0	0	0	0	0	0
schelp10	0.45	3	0	0	0	0	0	0	0	0

Wyniki analizy pokazują, że dane są dobrze dopasowane również do założenia podwójnej monotoniczności.

Globalny współczynnik skalowania:

```
H.hdiff = coefH(D.hdiff)
```

Globalny współczynnik skalowania H wyniósł 0,413 z 95% przedziałem ufności równym [0.33068, 0.49532]. Można to uznać za satysfakcjonującą dobroć dopasowania.

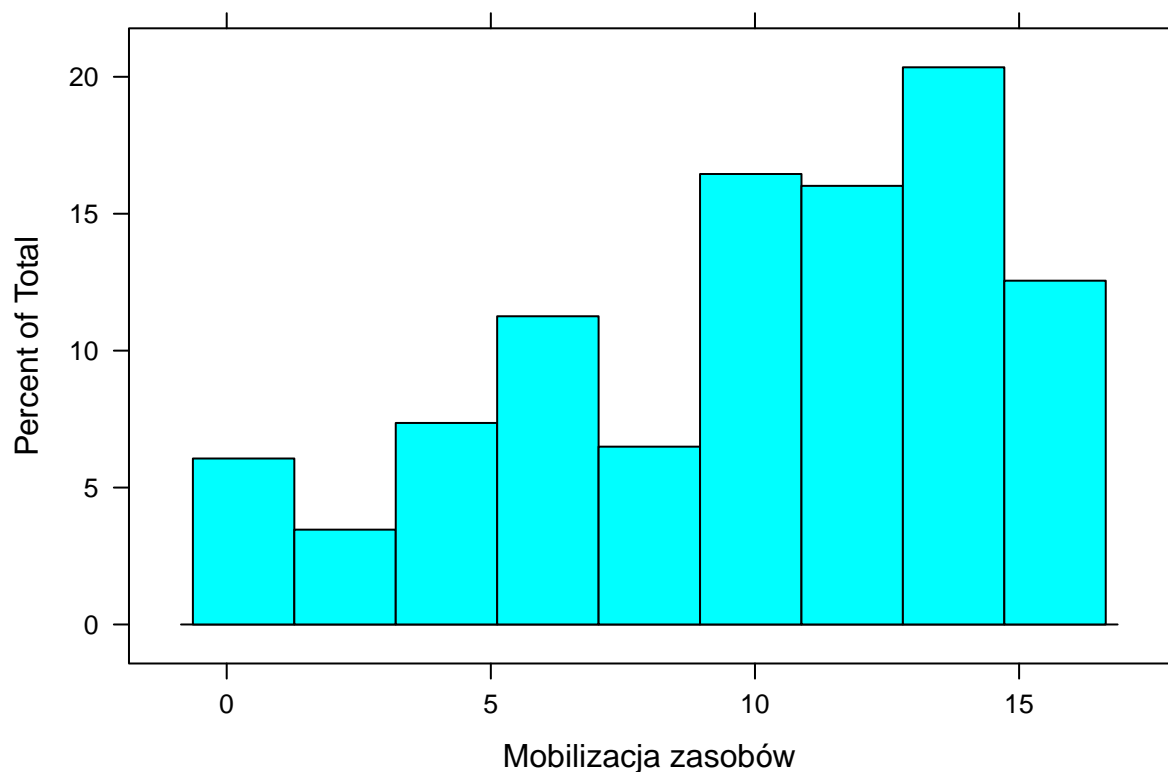
Rzetelność:

```
Rel.hdiff = check.reliability(D.hdiff, LCRC = TRUE)
```

Rzetelność mierzona współczynnikiem α -Cronbacha wyniosła 0,71. Metoda LCRC dała taki sam wynik. Oznacza to, że skala ma wystarczający poziom precyzji pomiaru, chociaż zdecydowanie nie jest on idealny.

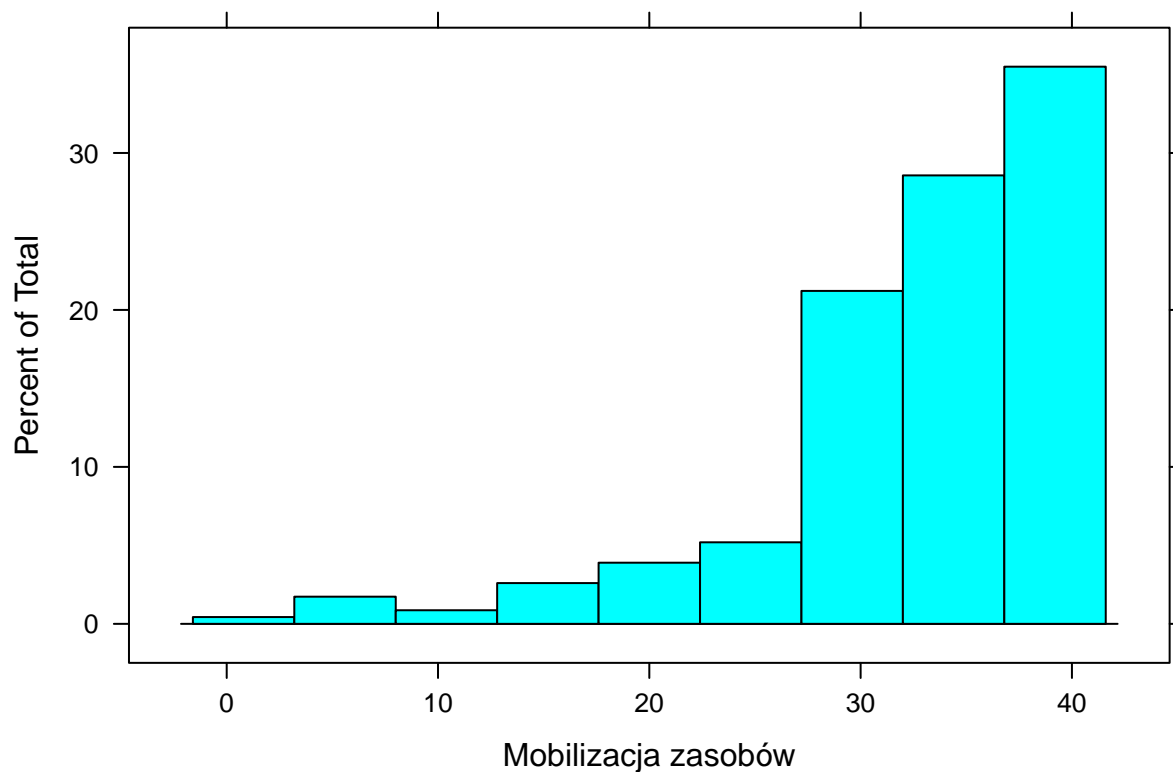
Jednakże bezsprzeczną przewagę tego podejścia pokazuje prosty histogram:

```
resmob = apply(D.hdiff, 1, sum)
histogram(resmob, xlab="Mobilizacja zasobów")
```



Jak widać jest on o wiele mniej lewoskośny od rozkładu poprzedniej skali:

```
histogram(~ resmob, data=D10, xlab="Mobilizacja zasobów")
```



Skalowanie kontaktów społecznych

Wybór i ograniczenie zbioru zmiennych tylko do tych, których średnie są mniejsze bądź równe 3:

```
D.kdiff = D.back[, grep("scknow", names(D.back))]
diff.know = apply(D.kdiff, 2, mean)
D.kdiff = D.kdiff[, names(diff.know)[diff.know <= 3]]
```

Skalowanie

Automatyczny dobór pozycji o wystarczająco dobrych współczynnikach skalowania w oparciu algorytm genetyczny:

```
scale.ga.kdiff = aisp(D.kdiff, search="ga", lowerbound=.35)
D.kdiff = D.kdiff[, rownames(scale.ga.kdiff)[scale.ga.kdiff[,1]==1]]
```

Wybrane pytania odnoszą się do znajomości osoby, która: pracuje w redakcji radia/gazety/telewizji itp. (2); jest muzkiem / dj'em (9); zna osobiście osoby rozpoznawalne z gazet/telewizji (14); prowadzi firmę, która zatrudnia pracowników (13). Bezsprzeczną zaletą nowego podziału na skale jest to, że skala druga ewidentnie odnosi się do innego wymiaru kapitału społecznego. Mobilizacja zasobów dotyczy przede wszystkim możliwości otrzymania konkretnej i dosyć specyficznej pomocy, podczas gdy skala kontaktów społecznych odnosi się do miększego rodzaju kapitału, który jest związany przede wszystkim ze znajomością specyficznych i/lub popularnych/prestiżowych osób.

Homogeniczna monotoniczność:

```
mono.kdiff = check.monotonicity(D.kdiff)
kable(summary(mono.kdiff))
```

	ItemH	#ac	#vi	#vi/#ac	maxvi	sum	sum/#ac	zmax	#zsig	crit
scknow3	0.49	4	0	0	0	0	0	0	0	0
scknow5	0.41	4	0	0	0	0	0	0	0	0
scknow13	0.39	4	0	0	0	0	0	0	0	0
scknow14	0.40	4	0	0	0	0	0	0	0	0

Dane są wysoce zgodne z modelem homogenicznej monotoniczności.

Podwójna monotoniczność:

```
io.kdiff = check.io(D.kdiff)
kable(summary(io.kdiff)$item.summary)
```

	ItemH	#ac	#vi	#vi/#ac	maxvi	sum	sum/#ac	tmax	#tsig	crit
scknow13	0.39	3	0	0	0	0	0	0	0	0
scknow5	0.41	3	0	0	0	0	0	0	0	0
scknow3	0.50	3	0	0	0	0	0	0	0	0
scknow14	0.40	3	0	0	0	0	0	0	0	0

Widać, że występuje również pełna zgodność z modelem podwójnej homogeniczności.

Globalny współczynnik skalowania:

```
H.kdiff = coefH(D.kdiff)
```

Globalny współczynnik skalowania wyniósł 0,374 z 95% przedziałem ufności równym [0.2956, 0.4524]. Jest to wartość akceptowalna, chociaż dolna granica przedziału minimalnie poniżej 0,30 wskazuje, że skala nie jest zbyt mocna.

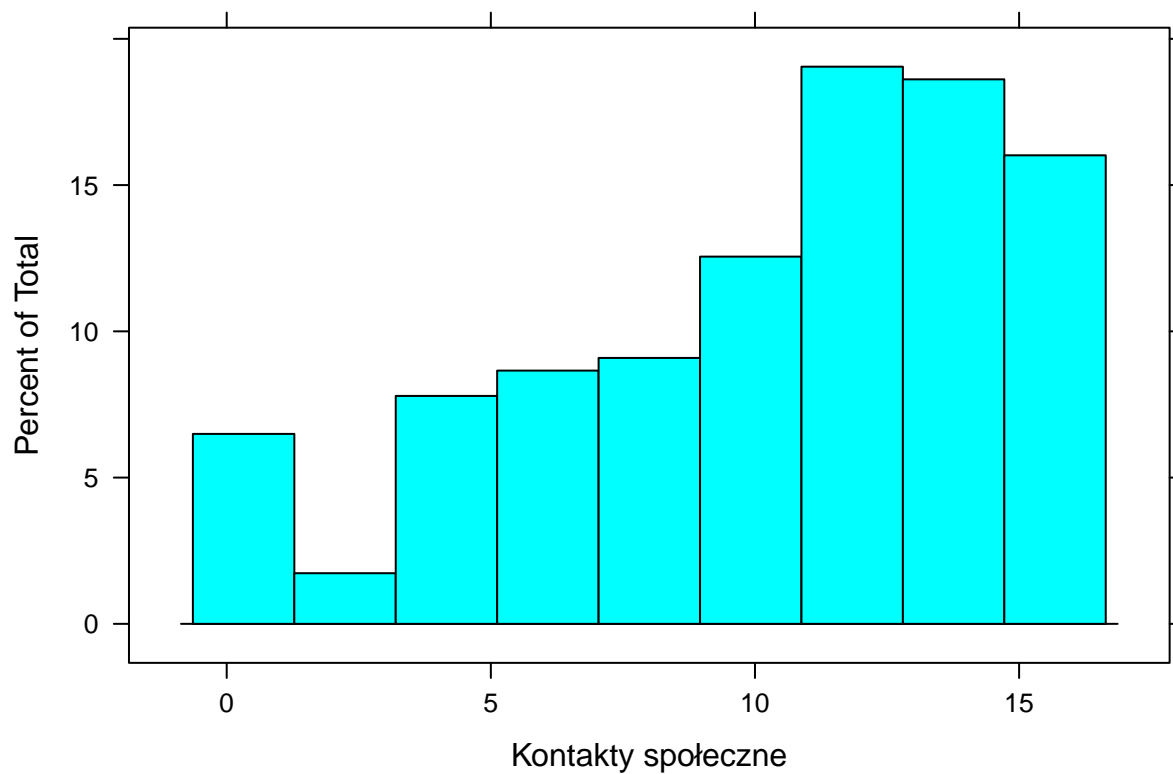
Rzetelność:

```
Rel.kdiff = check.reliability(D.kdiff, LCRC = TRUE)
```

Nienajlepszą jakość skali pokazują również wartości współczynników rzetelności. α -Cronbacha wyniosła 0,68 a współczynnik LCRC 0,66. Obie wartości można uznać za jeszcze akceptowalne, ale ewidentnie nie są one szczególnie dobre.

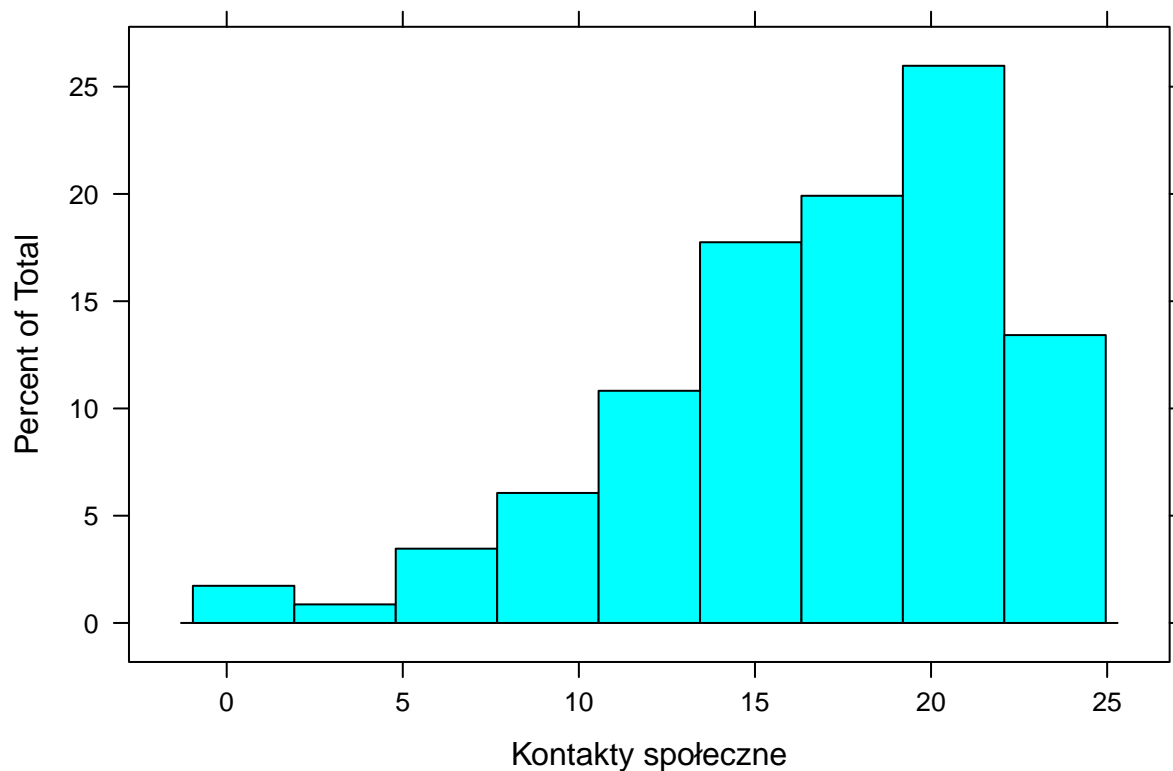
Niemniej jednak skala ta ma wielką zaletę, jaką jest lepszy potencjał dyskryminacyjny, co widać po jej rozkładzie:

```
soccont = apply(D.kdiff, 1, sum)
histogram(soccont, xlab="Kontakty społeczne")
```



Jest on o wiele bardziej symetryczny niż rozkład pierwszej wersji skali:

```
histogram(~ soccont, data=D10, xlab="Kontakty społeczne")
```



Dlatego też nowe wersje skali zostaną użyte w dalszych analizach. Jest to podyktowane przede wszystkim

tym, że będą one użyte w analizach opartych na współczynnikach korelacji, a jak wiadomo ucięte rozkłady i ograniczenia wariancji zmiennych bardzo silnie zaniżają ich wartości.

Koniec raporty