

Zadania do kursu R – Ankiety SUS

Jacek Matulewski

Wersja: 29 grudnia 2021

Najlepiej wykonywać w notebooku R Studio.

Dane wejściowe to zestaw trzynastu wypełnionych ankiet oceniających mobilną grę terapeutyczną. Ankieta składa się z pytań dotyczących treści gry, ankiety SUS¹ pozwalającej na ocenę użyteczności oraz trzy pytania otwarte. W tej analizie skupimy się jedynie na kwestionariuszu SUS, składającego się z dziesięciu stwierdzeń opisujących różne wrażenia dotyczące korzystania z gry. Każde stwierdzenie ma odpowiednik w postaci zanegowanej. Badany zaznacza na pięciostopniowej skali Likerta to, na ile identyfikuje się z poszczególnymi stwierdzeniami kwestionariusza (1 – Nie, 3 – Średnio, 5 – Tak).

Stwierdzenia ankiety SUS zostały przetłumaczone na język polski i dostosowane do tego, że oceniana była gra. Zmieniono zatem słowo „system” na „gra”, jak również zmieniona została kolejność niektórych pytań, szczególnie w przypadku, w którym pytanie i jego negacja występowały w oryginalne obok siebie (np. pytania 8 i 9). Poniżej w nawiasach przy wersji angielskiej podano oryginalny numer pytania.

1. Chciałbym grać w tę grę częściej.
(1) I think that I would like to use this system frequently.
Gra była niepotrzebnie trudna do obsługi (skomplikowana).
(2) I found the system unnecessarily complex.
Granie w grę nie sprawiało mi problemów (było wygodne).
(9) I felt very confident using the system.
2. Potrzebna była mi pomoc (techniczna), żeby rozpocząć granie.
(4) I think that I would need the support of a technical person to be able to use this system.
3. Poszczególne elementy gry tworzyły spójną całość.
(5) I found the various functions in this system were well integrated.
4. Uważam, że każdy może nauczyć się grać w tę grę bardzo szybko.
(7) I would imagine that most people would learn to use this system very quickly.
5. Gra była bardzo niewygodna w użyciu.
(8) I found the system very cumbersome to use.
6. Gra była prosta w użyciu.
(3) I thought the system was easy to use.
7. Poszczególne elementy gry były niedopasowane.
(6) I thought there was too much inconsistency in this system.
8. Gra wymagała ode mnie nauki obsługi smartfonu.
(10) I needed to learn a lot of things before I could get going with this system.

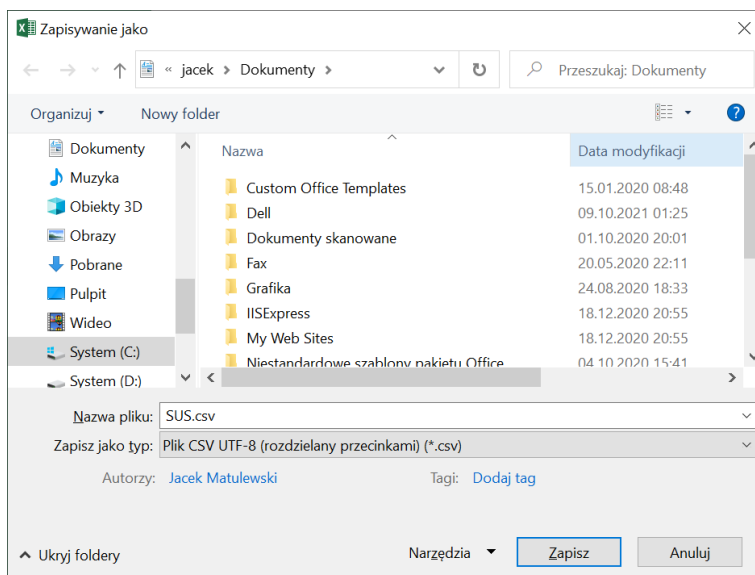
¹ Zob. <https://measuringu.com/sus/>, <https://www.usability.gov/how-to-and-tools/methods/system-usability-scale.html>, <https://www.surveylab.com/pl/blog/skala-uzytecznosc-systemu-sus/>, <https://doi.org/10.1080/10447310802205776>, <https://doi.org/10.1201/9781498710411>.

Zacznijmy od wprowadzenia danych do pliku CSV rozdzielanego średnikami. Ja zrobiłem to używając *Excela*, ale można to zrobić także za pomocą zwykłego notatnika systemu Windows. W kolumnach umieszczam numer badanego (nie będzie miał większego znaczenia) oraz liczby odpowiadające zaznaczonym na skali odpowiedzi (od 1 dla „Nie” do 5 dla „Tak”).

Zwróćmy uwagę, że ankieta użytkownika nr 3 zawiera odpowiedzi wprowadzone niezgodnie z instrukcją. Ten rekord zostanie później pominięty.

Lp	1	2	3	4	5	6	7	8	9	10		
1	1	1	1	5	1	5	5	1	4	3	1	
2	2	5	2	5	1	4	5	1	5	1	1	
3	3											
4	4	3	1	5	1	3	3	1	5	3	1	
5	5	3	1	3	1	3	4	3	3	3	3	
6	6	4	1	5	4	4	5	1	5	1	1	
7	7	2	2	4	2	3	5	1	4	2	1	
8	8	1	2	1	1	5	1	5	4	1	1	
9	9	5	2	4	1	3	5	5	1	3	1	
10	10	3	1	5	1	2	5	1	5	1	1	
11	11	1	1	1	1	1	5	5	5	5	1	
12	12	1	2	4	5	3	3	5	3	3	1	
13	13	2	1	4	1	4	4	1	4	4	1	
14												
15												
16												

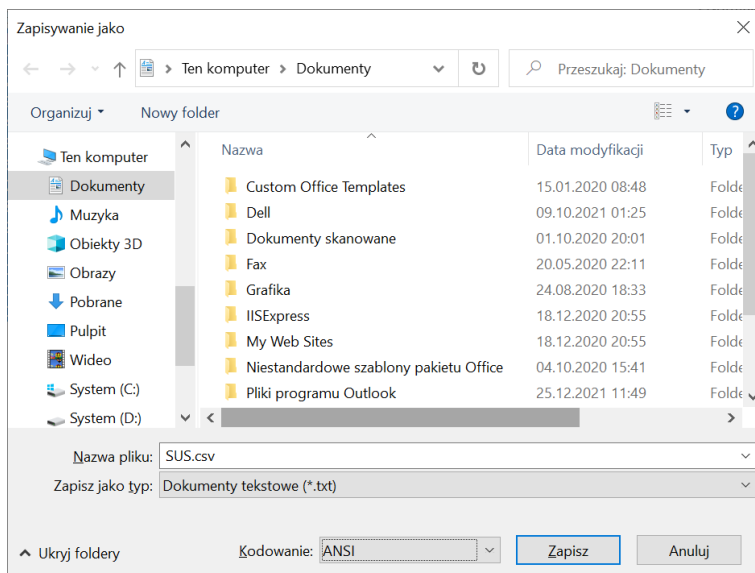
Zapisuję dane w pliku CSV UTF-8. Opis pliku w oknie dialogowym pokazuje, że będzie rozdzielany przecinkami, ale tak naprawdę będzie rozdzielany średnikami (dla formatu odpowiedniego dla naszego kraju przecinki kolidowałyby ze znakiem dziesiętnym, który akurat w tym pliku nie jest obecny).



Zapisany plik powinien wyglądać następująco:

```
Lp;1;2;3;4;5;6;7;8;9;10  
1;1;1;5;1;5;5;1;4;3;1  
2;5;2;5;1;4;5;1;5;1;1  
3;;;;;;;;;  
4;3;1;5;1;3;3;1;5;3;1  
5;3;1;3;1;3;4;3;3;3;3  
6;4;1;5;4;4;5;1;5;1;1  
7;2;2;4;2;3;5;1;4;2;1  
8;1;2;1;1;5;1;5;4;1;1  
9;5;2;4;1;3;5;5;1;3;1  
10;3;1;5;1;2;5;1;5;1;1  
11;1;1;1;1;1;5;5;5;5;1  
12;1;2;4;5;3;3;5;3;3;1  
13;2;1;4;1;4;4;1;4;4;1
```

Uwaga! Aby oszczędzić sobie kłopotu z kodowaniami znaków można otworzyć zapisany plik w *Notatniku* systemu Windows i zapisać go z kodowaniem ANSI. Dalej nie będę jednak zakładał, że format pliku został tak zmieniony.



Przejdźmy wreszcie do *R Studio*. Wczytajmy plik i przearanżujmy kolejność kolumn tak, aby odpowiadała kolejności z oryginalnego kwestionariusza w wersji angielskiej. Dzięki temu kod obliczający wyniki będzie mógł być bez modyfikacji użyty w przypadku wyników dla oryginalnego kwestionariusza.

```
dane <- read.csv("c:\\Users\\jacek\\Documents\\SUS.csv",  
               header = TRUE, sep = ";", row.names = 1,  
               fileEncoding = "UTF-8-BOM")  
dane <- dane[-c(3),] //usunięcie wyników dla trzeciego uczestnika  
dane <- dane[,c(1,2,8,4,5,9,6,7,3,10)]  
               //przywrócenie oryginalnej kolejności kolumn
```

Następnie przygotujemy funkcję, która przyjmie wektor (wiersz/rekord z ramki danych) lub całą ramkę i na tej podstawie obliczy wyniki SUS. Zaznaczone wartości dla pytań pozytywnych są przesuwane do skali od 0 do 4 ($n \rightarrow n-1$), a dla stwierdzeń negatywnych „odwracane” ($n \rightarrow 5-n$). Następnie tak obliczone punkty należy zsumować. W efekcie w przypadku stwierdzeń zawsze najbardziej przychylnych ocenianemu systemowi maksymalna suma byłaby równa 40, a przy konsekwentnie nieprzychylnych odpowiedziach równałaby się zeru. Aby uzyskać wynik w skali od 0 do 100 można pomnożyć sumę punktów przez 2.5. Nie można jednak tak przeskalowanego wyniku traktować jako „procentu użyteczności”. Przy wyznaczaniu ostatecznej oceny należy bowiem uwzględnić częstość uzyskiwania poszczególnych wyników w badaniach z użyciem SUS (por. pojęcie percentyli/centyli). Z podobnego powodu funkcja przeliczająca liczbę punktów uzyskanych na egzaminie na ocenę nie musi być liniowa (zaliczenia zaczynają się zwykle dopiero od 50% punktów).

```
obliczWynikSUS <- function(wiersz)
{
  plusy <-
    (wiersz[1]-1) +
    (wiersz[3]-1) +
    (wiersz[5]-1) +
    (wiersz[7]-1) +
    (wiersz[9]-1)
  minusy <-
    (5 - wiersz[2]) +
    (5 - wiersz[4]) +
    (5 - wiersz[6]) +
    (5 - wiersz[8]) +
    (5 - wiersz[10])
  wynik <- 2.5*(plusy + minusy)
  return(wynik)
}
```

Możemy łatwo przetestować funkcję korzystając z wektorów zawierających pozytywne, negatywne i pośrednie wartości:

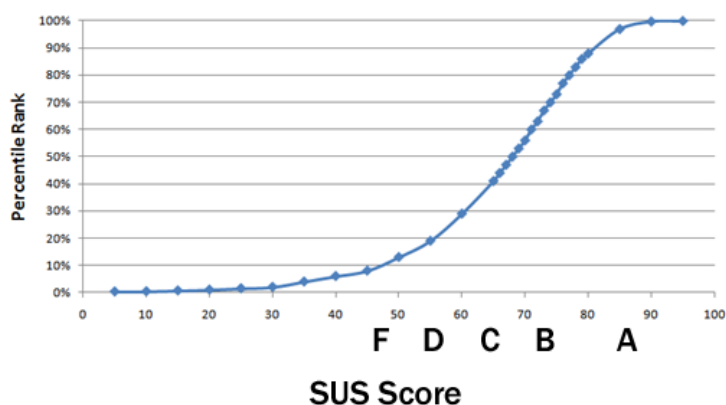
```
obliczWynikSUS(c(5, 1, 5, 1, 5, 1, 5, 1, 5, 1)) #max
obliczWynikSUS(c(1, 5, 1, 5, 1, 5, 1, 5, 1, 5)) #min
obliczWynikSUS(c(3, 3, 3, 3, 3, 3, 3, 3, 3, 3))
```

Użyjmy teraz funkcji na całej ramce danych i zapiszmy wynik do wektora, a następnie obliczmy wynik średni i odchylenie standardowe:

```
wynikiSUS <- obliczWynikSUS(dane)
wynikiSUS <- wynikiSUS[,1] //zmiana na wektor

mean(wynikiSUS) //71.66667 pkt
sd(wynikiSUS) //16.07275 pkt
```

Jak zazaczyłem wyżej uzyskanie wyniku 50 nie oznacza, że aplikacja jest użyteczna w 50%. Wyniki są zwykle przeliczane do pięciostopniowej skali ocen używanych w szkolnictwie amerykańskim: F, D, C, B i A. W tabeli 1 przedstawiono zakresy punktów kwalifikujące do poszczególnych ocen.



Rysunek 1. Centyle ocen dla ankiety SUS i wynikające z nich zakresy finalnych ocen
(Źródło: <https://measuringu.com/sus/>)

Tabela 1. Przeliczanie punktów uzyskanych w ankiecie SUS na oceny

Ocena	Punkty
F	0-50
D	51-66
C	67-68
B	69-80
A	81-100

Funkcja obliczająca finalną ocenę może zatem wyglądać tak, jak na listingu poniżej. W ostatniej linii widoczna jest pętla testująca działanie funkcji.

```
wyznaczOcenęSUS <- function(punkty)
{
  wynik <- "F"
  if(punkty > 50) wynik <- "D"
  if(punkty > 66) wynik <- "C"
  if(punkty > 68) wynik <- "B"
  if(punkty > 80) wynik <- "A"
  return(wynik)
}

for(i in c(1:100)) print(paste(i,wyznaczOcenęSUS(i))) //test
```

Funkcji możemy użyć na ostatecznym wyniku (pojedynczej liczbie):

```
wyznaczOcenęSUS(mean(wynikiSUS)) //ocena B
```

Możemy jednak tę funkcję napisać nieco inaczej - tak, żeby można jej było użyć także na całym wektorze liczb:

```
wyznaczOcenęSUS <- function(punkty)
{
  wynik <- ifelse(punkty < 51, "F",
                 ifelse(punkty < 67, "D",
                       ifelse(punkty < 69, "C",
                             ifelse(punkty < 81, "B", "A"))))
  return(wynik)
}
```

Wówczas nadal można jej użyć dla wyniku uśrednionego dla całej grupy:

```
mean(wynikiSUS) //71.66667 pkt
wyznaczOcenęSUS(mean(wynikiSUS)) //ocena B
```

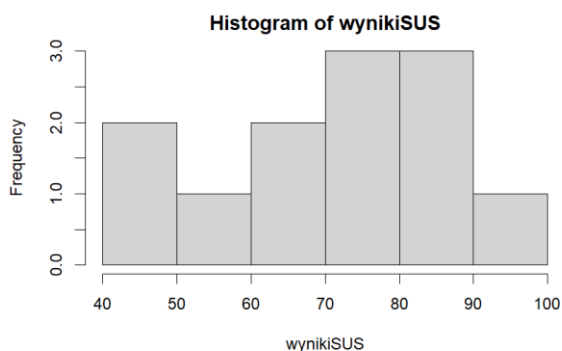
Można jej jednak również użyć dla wektora punktów przyznanych przez poszczególne osoby, co pozwoli na policzenie rozkładu ocen, który można zaprezentować na wykresie.

```
ocenySUS <- wyznaczOcenęSUS(wynikiSUS)
rozkładOcenSUS <- rev(table(ocenySUS))
barplot(rozkładOcenSUS, xlab = "Oceny", ylab = "Liczba")
```



Można oczywiście także wyświetlić rozkład oryginalnych punktów:

```
wynikiSUS
hist(wynikiSUS)
```



Dodatek

To jednak nie koniec. W ostatnich publikacjach na temat SUS² uzasadnia się możliwość wydzielenia z ostatecznego wyniku użyteczności (ang. *usability*) podskali łatwości nauki systemu (ang. *learnability*). Składają się na nią wyniki dwóch stwierdzeń (4 i 10) oryginalnego kwestionariusza SUS.

Poniższa funkcja oblicza odpowiednią wartość, ponownie przeskalowaną do zakresu od 0 do 100:

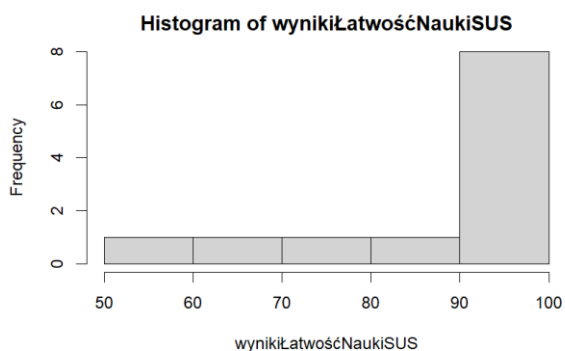
```
obliczŁatwośćNaukiSUS <- function(wiersz)
{
  wynik <-
    (5 - wiersz[4]) +
    (5 - wiersz[10])
  return(12.5 * wynik)
}

obliczŁatwośćNaukiSUS(c(5, 1, 5, 1, 5, 1, 5, 1, 5, 1)) #max
obliczŁatwośćNaukiSUS(c(1, 5, 1, 5, 1, 5, 1, 5, 1, 5)) #min
obliczŁatwośćNaukiSUS(c(3, 3, 3, 3, 3, 3, 3, 3, 3, 3))
```

Korzystając z tej funkcji możemy ponownie przeprowadzić obliczenia dla całego zestawu danych:

```
wynikiŁatwośćNaukiSUS <- obliczŁatwośćNaukiSUS(dane)
wynikiŁatwośćNaukiSUS <- wynikiŁatwośćNaukiSUS[,1]

mean(wynikiŁatwośćNaukiSUS)
table(wynikiŁatwośćNaukiSUS)
hist(wynikiŁatwośćNaukiSUS)
```



² <https://doi.org/10.1007/s10339-009-0268-9>