

4. TRANSFORMACE PROMĚNNÝCH

Proměnné lze do datové matice zadávat ručně (např. vyvoláním editoru datové matice pomocí příkazu EDIT), data lze importovat z různých formátů (viz příkaz IMPORT a USESPSS). Existují ale i speciální příkazy, které umožňují s proměnnými pracovat v rámci syntaxí.

4.1 Popisování proměnných

Stata rozlišuje několik názvů (name) a popisků (label) s následujícím významem:

Variable name: název proměnné ve formátu, ve kterém ho musíme zadávat při používání všech příkazů Staty. Nesmí se lišit ani ve velikosti písmen, ani v mezerách. Musí být zkrátka zadán přesně tak, jak ho Stata zná.

Variable label: popisek proměnné. Slouží pouze pro lepší orientaci uživatele, Stata s ním nijak nepracuje, jen ho zobrazuje při výpisu výsledků namísto jména proměnné.

Value label: popisek hodnot proměnné. Pro možnost využívat statistické procedury jsou odpovědi respondentů tzv. kódovány do číselné podoby (např. pohlaví muž = 1, žena = 2, vzdělání ZŠ = 1, SŠ = 2, VŠ = 3 apod.). Pro snazší orientaci uživatele je možno přiřadit jednotlivým číselným hodnotám popisky (např. hodnotě 1 přiřadíme popisek „muž“). Tyto popisky se nikam nezadávají, Stata s nimi nepracuje, jen je zobrazuje při výpisu výsledků.

Tip: value label se často opakují (např. odpovědi na otázky Likertovy škály „zcela nesouhlasím“, „částečně nesouhlasím“, „ani nesouhlasím, ani souhlasím“, „částečně souhlasím“, „zcela souhlasím“). Proto Stata pracuje s popisky hodnot proměnných tak, že vytvoří pojmenovaný soubor popisků (např. výše uvedená škála by se mohla jmenovat LIKERTOVA_SKALA_POPISKY). Tento soubor popisků pak můžeme přiřadit několika proměnným.

```
. label variable kat_nejvzds "Kategorie nejvyššího dosaženého vzdělání"
. label define KAT_NEJVZDS 1 "bez maturity" 2 "s maturitou" 3 "VŠ"
. label values kat_nejvzds KAT_NEJVZDS
. tab kat_nejvzds
```

Kategorie nejvyššího dosaženého vzdělání	Freq.	Percent	Cum.
bez maturity	6,645	50.93	50.93
s maturitou	3,933	30.14	81.08
VŠ	2,469	18.92	100.00
Total	13,047	100.00	

Obrázek 1 Ukázka práce s popisky proměnných a popisky hodnot

Příklad: proměnná POHL (variable name) bude mít variable label např. „Pohlaví respondenta či respondentky“, možné hodnoty 1 a 2 s popisky (variable label) 1 = muž, 2 = žena. Popisek hodnot pojmenujeme například POPISKY_POHLAVI. Stejný popis hodnot pak můžeme použít pro proměnnou POHL_DITE_1 (Pohlavní 1. dítěte) nebo POHL_NEJ_OS_DOM (Pohlaví nejstarší osoby žijící ve stejné domácnosti).

LABEL – příkaz pro práci s popisky proměnných a popisky hodnot proměnných. Používá se pro vytvoření popisku proměnné (label variable), definici souborů s popisky hodnot (label define) a přiřazení souboru s popisky hodnot konkrétní proměnné (label values). Pro výpis informací o popiscích hodnot slouží label list.

```
label variable kat_nejvzds "Kategorie nejvyššího dosaženého vzdělání"
label define KAT_NEJVZDS 1 "bez maturity" 2 "s maturitou" 3 "VŠ"
label values kat_nejvzds KAT_NEJVZDS
```

4.2 Mazání proměnných a respondentů

Pro práci s jednotlivými případy a jednotlivými proměnnými má Stata dva komplementární příkazy: KEEP a DROP. Jejich výsledek je stejný, záleží proto jen na uživateli, který z nich v konkrétním případě použije.

DROP – smaže proměnnou nebo proměnné zadané za příkazem DROP.

```
drop age
drop age mstat estat
```

KEEP – ponechá proměnnou nebo proměnné zadané za příkazem KEEP, ostatní smaže.

```
keep age mstat estat
```

DROP – smaže případy splňující zadanou podmínku.

```
drop if age>=30
```

KEEP – ponechá případy splňující zadanou podmínku, ostatní smaže.

```
keep if age>43
```

4.3 Změna proměnných a jejich obsahu

RENAME – změní jméno proměnné. Na prvním místě se za příkazem RENAME uvádí aktuální jméno, na druhém místě se uvádí nové jméno. Například následující příklad znamená „přejmenuj proměnnou, která se aktuálně jmenuje age; nové jméno nechť je vek“.

```
rename age vek
```

REPLACE – změní obsah proměnné (výpočtem). Příkaz se používá ve tvaru REPLACE nazev_promenne = vypocet. Příkaz REPLACE je možno kombinovat s podmínkami IF.

```
replace vek=2016-birthy
replace vek=60 if vek>60
```

RECODE – změní obsah proměnné (rekódováním). Po názvu proměnné nebo proměnných se uvádí přepisovací pravidla, která stanoví, která hodnota proměnné se přepíše na kterou novou hodnotu. Zajímavými parametry jsou COPYREST, který zkopíruje zbývající hodnoty, pro něž jsme nestanovili přepisovací pravidla, a GENERATE, který rekóduje výsledky do nové proměnné (viz dále). Pravidla se zadávají jednoduše pomocí znaménka rovná se (1 = 2) (2 = 3) (3 = 1). Rozsah „od do“ se zadává pomocí lomítka (1/3 znamená „od jedné do tří včetně“).

```
recode vek 0/15=1 16/65=2 66/110=3
recode vek 0 1 2 3 4 5 6 7 8 9 10 = 1, copyrest
recode vek 0/15=1 if mstat>1
recode gearn grespo (1=5) (2=4) (3=3) (4=1) (5=1) (88 99 = .)
```

```
. drop vek
. rename age vek
. replace vek = 2016-birthy
(13034 real changes made, 2 to missing)
. recode vek (0/15=1) (16/65 = 2) (66/96 = 3), generate (vek_kat)
(12955 differences between vek and vek_kat)
```

Obrázek 2 Ukázka práce s proměnnými

4.4 Vytváření nových proměnných

GENERATE – vytvoří novou proměnnou na základě výpočtu ze staré proměnné. Lze použít i jako parametr u příkazu recode. Před názvem nové proměnné můžete zadat její datový typ (byte, int, long, float, double).

```
generate vek2=vek*vek
generate int vek2=vek*vek
generate vek2=vek*vek if sex ==1
recode vek 0/15=1 16/34=2 35/100=3, generate (vek2)
```

EGEN – rozšířené vytváření nových proměnných. Tento příkaz je schopen vygenerovat novou proměnnou s využitím většího množství funkcí. Mezi zajímavé možnosti patří například funkce MIN, MAX, MEAN a MEDIAN, které generují novou proměnnou, jejíž všechny případy nabývají hodnot minima, maxima, průměru či mediánu jiné proměnné. Další možnosti skýtají funkce DIFF (vrací jedničku, pokud se hodnoty dvou proměnných v daném případě liší) nebo RANK (vrací pořadí konkrétní hodnoty v případě, kdy proměnnou seřadíte).

```
egen nej_vel = max (vek)
egen prum_vek = mean (vek)
egen rozdil = diff (gearn grespo)
egen poradi_veku = rank (vek)
```