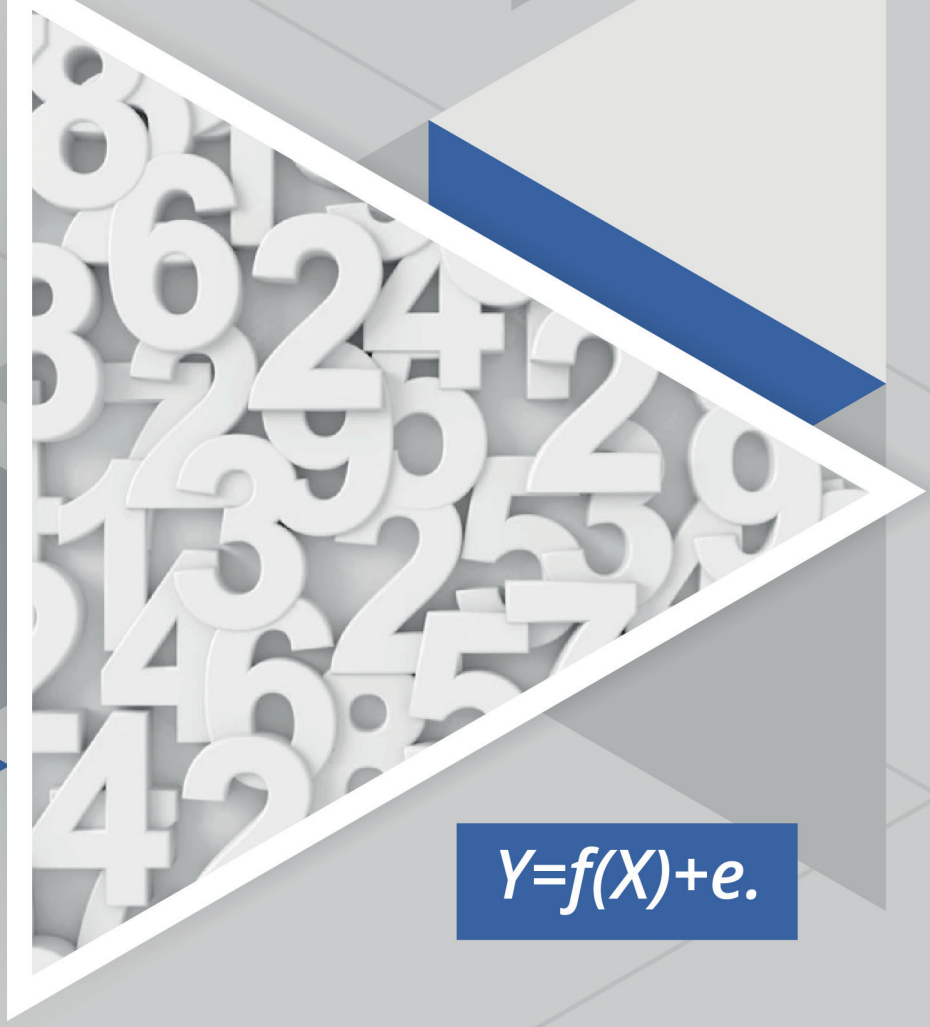
OSNOVE KORIŠTENJA
PROGRAMA STATA



$$Y=f(X)+e.$$

UNAPREĐENJE ISTRAŽIVANJA TRŽIŠTA RADA

Finansira Evropska unija

Naziv projekta:	„Jačanje kapaciteta institucija tržišta rada poboljšanjem metodologije istraživanja tržišta rada“
Ref. br projekta:	EuropeAid / 140152 / DH / SER / BA
Datum potpisa ugovora:	3. jul 2020. godine
Broj ugovora:	2020 / 417-153
Početak projekta:	1. septembar 2020.
Adresa:	La Benevolencija 8, 71000 Sarajevo, BiH
Naslov priručnika:	Priručnik za prvi nivo edukacije o korištenju softvera Stata
Autor:	Nermin Oruč
Datum:	22. februar 2022. godine

Ovaj materijal izrađen je uz tehničku podršku projekta „Jačanje kapaciteta institucija tržišta rada unapređivanjem metodologije istraživanja tržišta rada“, koji finansira Evropska unija i provodi konzorcijum NIRAS IC Sp z oo, GOPA Worldwide Consultants, GOPA mbH Njemačka i Zavod za zapošljavanje Republike Francuske.

Sadržaj ove publikacije isključiva je odgovornost autora i ne odražava nužno stavove Evropske unije.

SADRŽAJ

1. Uvod

Priručnik za korištenje softvera Stata izrađen je za polaznike 5. modula edukacije Akademije za unapređenje istraživanja tržišta rada pod nazivom „Osnove korištenja programa Stata u istraživanjima tržišta rada“ organizirane u okviru projekta Evropske unije „Unapređenje istraživanja tržišta rada“.

Svrha mu je da čitaocima obezbijedi polazna znanja iz korištenja softvera Stata, a kako bi mogli samostalno analizirati prikupljene podatke koje dobijaju. Ovaj Priručnik može poslužiti polaznicima naših edukacija kao kratak podsjetnik na komande koje smo koristili u okviru edukacija budući da sadrži objašnjenja načina na koje mogu upotrebljavati analizu i istu prilagođavati prema svojim potrebama.

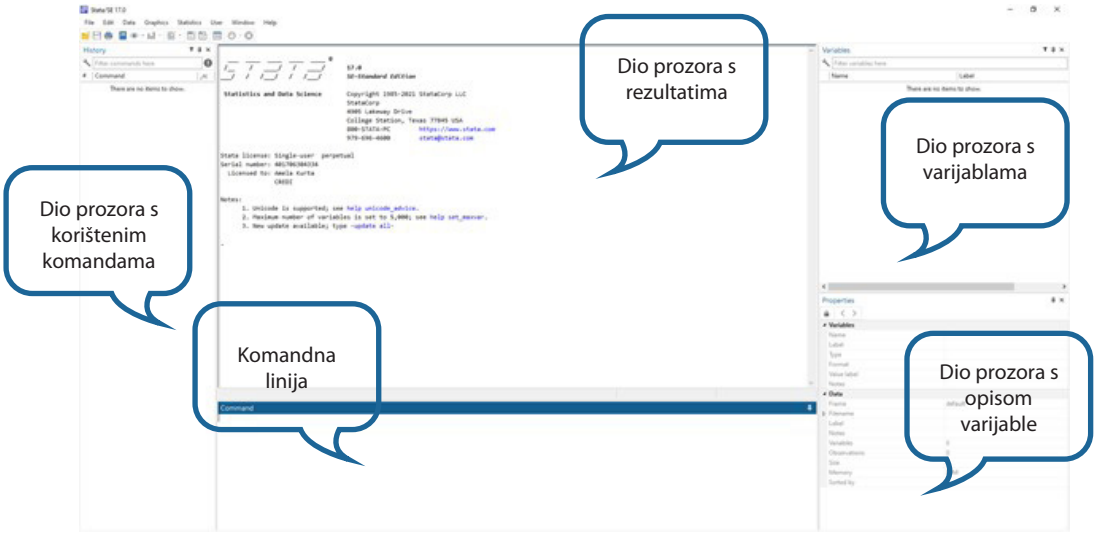
Na početku Priručnika ćemo ih upoznati s osnovnim elementima prozora Stata, te načinima preuzimanja ili učitavanja u Stata prozor podataka snimljenih u različitim formatima. U drugom dijelu Priručnika ćemo ih upoznati s uobičajenim komandama koje se koriste za manipulaciju podacima, kao i za označavanje varijabli i opservacija, a kako bi bazu podataka pripremili za dalji rad i ispis rezultata koje mogu direktno koristiti u izvještajima o istraživanjima. Nakon što ovladaju osnovnim vještinama rada s bazom podataka, varijablama i opservacijama, Priručnik ih u nastavku upoznaje s osnovnim komandama koje se koriste u opisivanju podataka, odnosno deskriptivnoj statistici, a koje uključuju komande poput: summarize, describe, tabulate i tabstat. Podaci se mogu opisivati i prikazivati i korištenjem grafičkih prikaza, pa se ovim Priručnikom pojašnjavaju i osnovne komande koje se mogu koristiti za iscrtavanje različitih vrsta grafikona. Nakon upoznavanja s podacima koji se nalaze u njihovoj bazi podataka, ovaj Priručnik će im pružiti i osnovna saznanja o korištenju korelacione i regresione analize, kao i osnovnih dijagnostičkih testova nakon provedene regresione analize. Na kraju Priručnika nalaze se i komande koje mogu koristiti za kreiranje uzoraka u bazama koje sadrže podatke za okvir uzorka ili kada žele izolovati određenu grupu u okviru prikupljenih podataka i analizu primijeniti samo na izabranom poduzorku.

Priručnik je izrađen tako da su za svaku komandu navedeni njen opis, generički oblik sintakse, kao i dodatne opcije koje se mogu koristiti uz određene komande. Uz svaku komandu naveden je po najmanje jedan praktičan primjer za koji su korišteni podaci iz baze auto.dta koja dolazi predinstalirana sa softverom Stata. Svim polaznicima se preporučuje da na edukaciju ponesu i koriste vlastite baze podataka, a kako bi pored sticanja praktičnih znanja i iskustva u toku edukacije mogli napraviti i .do fajlove za analizu podataka koji će im koristiti u budućem radu.

2. Prozor Stata

Prije predstavljanja izabranih komandi objasnit ćemo osnovne dijelove prozora Stata, jer ćemo u nastavku teksta upućivati na pojedine dijelove prozora.

Prozor programa Stata s označenim dijelovima prikazan je ispod (radi se o verziji za Windows; verzija za MacOS se neznatno razlikuje).



Dijelovi prozora su:

1. Dio prozora s rezultatima. U ovom dijelu prozora prikazuje se rezultat svake aktivnosti provedene u programu Stata.

2. Dio prozora s varijablama. Ovaj dio prozora sadrži listu varijabli sadržanih u bazi podataka koja se trenutno koristi za analizu.

3. Dio prozora s opisom varijable. Nakon klika na neku od varijabli u dijelu prozora s varijablama, njene osnovne karakteristike bit će prikazane u ovom dijelu.

4. Dio prozora s korištenim komandama. U ovom dijelu izlistavaju se sve prethodno korištene komande, tako da on služi za praćenje i provjeru samog procesa analize, kao i za brži unos sintakse u komandnu liniju klikom na prethodno korištenu komandu iz ovog dijela.

5. Komandna linija. U ovom dijelu upisuje se komanda kojom se programu naređuje provođenje određenog postupka.

3. Preuzimanje podataka

Ukoliko su podaci u formatu datoteke .dta, dovoljno je kliknuti na datoteku ili u prozoru Stata otići na opciju File>Open i naći odgovarajuću datoteku u folderu u kojem je prethodno sačuvana.

Ako se radi o formatima koji se mogu direktno učitati u program Stata (npr. xls, csv, txt, ...), potrebno je otići na opciju File>Import i izabrati odgovarajuću opciju (npr. Excel spreadsheet ako je datoteka u formatu .xls).

U našem slučaju, koristimo podatke koji se nalaze pohranjeni u memoriji programa Stata, tako da su na raspolaganju i mogu se koristiti u bilo kojoj situaciji. Ove podatke koristimo kako bi otišli na File>Example Datasets>Example datasets installed with Stata i pored odgovarajućeg skupa podataka (u našem slučaju auto.dta) kliknuli na poveznicu «use». Nakon klika, u glavnom prozoru programa Stata možemo vidjeti da je u dijelu s varijablama izlistan skup od 12 varijabli (make, price, mpg, ...), dok je u dijelu prozora za rezultate napisano:

```
. sysuse auto.dta

(1978 Automobile Data)
```

Prvi red označava komandu (koju sljedeći put možemo ukucati u komandnu liniju i dobiti isti rezultat, tj. unesene podatke za obradu), a drugi red prikazuje dodatne informacije, u ovom slučaju opis unesenih podataka (dakle, vidimo da se radi o podacima o automobilima iz 1978. godine).

Ako izaberete Data>Data Editor>Data Editor (Edit ili Browse), moći ćete vidjeti tabelu s podacima koje koristite u analizi. U slučaju učitavanja podataka iz formata .xls, u ovom prozoru biste trebali imati istovjetan prikaz podataka kao u datoteci .xls (s istim kolonama i redovima).

4. Komande za manipulaciju varijabli: generate i replace

Obično je nakon učitavanja podataka potrebno kreirati određene varijable korištenjem već postojećih podataka ili urediti postojeće varijable npr. korištenjem drugih kategorija podataka ili pretvorbom kontinuiranih varijabli u kategorijske varijable (npr. razvrstavanjem godina starosti u intervalne petogodišnje grupe). Ove dvije komande najčešće se koriste skupa i zbog toga ćemo ih zajedno objasniti.

4.1 Opis

generate kreira novu varijablu. Vrijednosti varijable specificirane su izrazom =exp. Ako tip varijable nije specificiran, tip nove varijable određen je rezultatom izraza =exp. Komanda se skraćeno piše gen.

replace mijenja sadržaj postojećih varijabli. S obzirom na to da ova komanda mijenja podatke, ne može se pisati skraćeno.

Nove i izmijenjene varijable dodaju se na samom kraju postojeće baze podataka, tako da ih možete pronaći na samom kraju liste u dijelu prozora s listom varijabli.

4.2 Sintaksa

Kreiranje nove varijable

generate nazivvarijable =exp if

Zamjena sadržaja postojećih varijabli

replace nazivvarijable=exp if

If kvalifikator koristi logične izraze da odredi koje opservacije će se koristiti. Ako je izraz tačan opservacija se koristi u komandi, inače se preskače.

Operatori čiji rezultati su ili tačni ili netačni su:

<	Manje od
<=	Manje ili jednako od
==	Jednako
>	Veće od
>=	Veće ili jednako od
!=	Nije jednako
&	I
|	Ili
!	Ne (logička negacija)

Korištenjem komande replace možete učiniti sve isto kao i korištenjem komande generate. Jedina razlika između komandi jeste to što replace zahtijeva da varijabla već postoji, dok generate zahtijeva novu varijablu. Zapravo, generate i replace imaju isti kod unutar programa Stata. Pošto je Stata interaktivan sistem, preporučujemo da razlikujete zamjenu postojećih vrijednosti i generisanje novih kako slučajno ne biste zamijenili vrijedne podatke dok razmišljate o kreiranju nove vrste informacija.

4.3 Primjer

Koristimo bazu podataka auto.dta.

Želimo kreirati novu varijablu mpg2=mpg+200.

. gen mpg

variable mpg already defined

r(110);

. gen mpg2=mpg+200

U prvom slučaju varijabla već postoji i u prozoru rezultata ispisuje se greška, s obzirom na to da ista varijabla ne može biti kreirana dva puta i moguće je samo mijenjati podatke unutar postojeće varijable. U drugom slučaju uspjeli smo kreirati novu varijablu. Nova varijabla dodana je popisu svih varijabli, a njen sadržaj možete pogledati u pregledniku (Data Editor/Browser) na samom kraju baze podataka.

. replace mpg2=0 if mpg>=20

(39 real changes made)

U ovom slučaju zamijenili smo već postojeće opservacije unutar varijable mpg2 s 0 pod uslovom da je vrijednost mpg>=20; ostale opservacije unutar varijable mpg2 nisu se promijenile. Moguće je i na ovaj način koristiti komandu replace:

. replace mpg2=200 if foreign==0

(52 real changes made)

Ovdje smo pređeni broj kilometara (mpg2) za sve automobile s domaćeg tržišta (foreign==0) odredili na nivou 200.

5. Komande za označavanje varijabli i opservacija: label var, label define, label values

5.1 Opis

Komanda **label data** dodaje oznaku (do 80 karaktera) bazi podataka učitanoj u memoriju. Oznake baze podataka prikazuju se kada se koristi ta baza podataka ili kada se koristi opcija za opis baze podataka (describe). Ako nikakva oznaka nije specificirana, sve postojeće oznake su uklonjene.

Komanda **label variable** dodaje oznaku (do 80 karaktera) varijabli. Ako oznaka nije specificirana, sve postojeće oznake varijabli su uklonjene.

Komanda **label define** definiše listu sa do 65,536 (1,000 za Small Stata) asocijacija cijelih brojeva i tekstualnih oznaka. Oznake vrijednosti pridružene su varijablama po vrijednostima označenog.

Komanda **label values** dodaje oznaku vrijednosti za grupu varijabli (varlist). Ako je . specificirana umjesto lblname, bilo koja postojeća oznaka vrijednosti je uklonjena s varliste. Oznaka vrijednosti, međutim, nije obrisana. Sintaksa label values varname (a da ništa ne prati varname) ponaša se isto kao i određivanje . (missing values). Oznake vrijednosti mogu sadržavati do 32,000 karaktera.

Komanda **label dir** izlistava sva imena oznaka vrijednosti pohranjenih u memoriji.

Komanda **label list** izlistava imena i sadržaj oznaka vrijednosti pohranjenih u memoriji.

Komanda **label copy** pravi kopiju već postojećih oznaka vrijednosti.

Komanda **label drop** eliminiše oznake vrijednosti.

Komanda **label save** spašava oznake vrijednosti u do. datoteku. Ovo je praktično za oznake vrijednosti koje nisu povezane s varijablama zato što se ove oznake ne spašavaju s podacima.

Od gore navedenih detaljnije ćemo objasniti label variable (skraćeno label var), label define i label values.

5.2 Sintaksa

Označavanje varijable

label variable nazivvarijable [“label”]

Definisanje oznake vrijednosti

label define Lnazivvarijable “label” [#”label”]

Dodavanje oznake vrijednosti varijablama

label values nazivvarijable Lnazivvarijable

5.3 Primjer

Koristimo bazu podataka auto.dta.

Prvo ćemo opisati trenutni izgled baze podataka na osnovu komande describe. Komandu describe posebno opisujemo u nastavku Priručnika, a ovdje je koristimo samo za praćenje izmjena koje činimo nad bazom podataka.

```
. describe

Contains data from C:\Program Files\Stata14\ado\base/a/auto.dta
  obs:            74                          1978 Automobile Data
 vars:            13                          13 Apr 2014 17:45
 size:         3,478                          (_dta has notes)
-------------------------------------------------------------------------------
              storage   display    value
variable name   type    format     label      variable label
-------------------------------------------------------------------------------
make            str18   %-18s                 Make and Model
price           int     %8.0gc                Price
mpg             int     %8.0g                 Mileage (mpg)
rep78           int     %8.0g                 Repair Record 1978
headroom        float   %6.1f                 Headroom (in.)
trunk           int     %8.0g                 Trunk space (cu. ft.)
weight          int     %8.0gc                Weight (lbs.)
length          int     %8.0g                 Length (in.)
turn            int     %8.0g                 Turn Circle (ft.)
displacement    int     %8.0g                 Displacement (cu. in.)
gear_ratio      float   %6.2f                 Gear Ratio
foreign         byte    %8.0g       origin    Car type
mpg2            float   %9.0g
-------------------------------------------------------------------------------
Sorted by: foreign
     Note: Dataset has changed since last saved.
```

Vidimo da naša nova varijabla mpg2 nema oznaku varijable niti oznake vrijednosti opservacija unutar varijable. Prvo ćemo odrediti naziv (label) ove varijable kao „Mileage 2“, pa ćemo ponovo prikazati opis sadržaja baze podataka.

Unosimo sljedeću sintaksu:

```
. label var mpg2 „Mileage 2“
```

U ispisu rezultata nakon komande describe vidimo sljedeći rezultat:

```
. label var mpg2 "Mileage 2"

. describe

Contains data from C:\Program Files\Stata14\ado\base/a/auto.dta
  obs:            74                          1978 Automobile Data
 vars:            13                          13 Apr 2014 17:45
 size:         3,478                          (_dta has notes)
-------------------------------------------------------------------------------
              storage   display    value
variable name   type    format     label      variable label
-------------------------------------------------------------------------------
make            str18   %-18s                 Make and Model
price           int     %8.0gc                Price
mpg             int     %8.0g                 Mileage (mpg)
rep78           int     %8.0g                 Repair Record 1978
headroom        float   %6.1f                 Headroom (in.)
trunk           int     %8.0g                 Trunk space (cu. ft.)
weight          int     %8.0gc                Weight (lbs.)
length          int     %8.0g                 Length (in.)
turn            int     %8.0g                 Turn Circle (ft.)
displacement    int     %8.0g                 Displacement (cu. in.)
gear_ratio      float   %6.2f                 Gear Ratio
foreign         byte    %8.0g       origin    Car type
mpg2            float   %9.0g                 Mileage 2
-------------------------------------------------------------------------------
Sorted by: foreign
     Note: Dataset has changed since last saved.
```

Iz gornjeg prikaza vidite da je naša varijabla mpg 2 dobila svoju oznaku „Mileage 2“. Sada ćemo definisati kako će se prikazivati sadržaj varijable mpg2. Odlučili smo da svi koji imaju vrijednost varijable 200 budu označeni kao „domaći“ i da svi koji imaju vrijednost 0 budu označeni kao „ostali“, pa ćemo ponovo opisati sadržaj baze podataka.

```
. label define lmpg2 200 “domaći” 0 “ostali”

. label values mpg2 lmpg2
```

Ispis rezultata nakon primjene komande describe je sljedeći:

```
Contains data from C:\Program Files\Stata14\ado\base/a/auto.dta
  obs:            74                          1978 Automobile Data
 vars:            13                          13 Apr 2014 17:45
 size:         3,478                          (_dta has notes)
-------------------------------------------------------------------------------
              storage   display    value
variable name   type    format     label      variable label
-------------------------------------------------------------------------------
make            str18   %-18s                 Make and Model
price           int     %8.0gc                Price
mpg             int     %8.0g                 Mileage (mpg)
rep78           int     %8.0g                 Repair Record 1978
headroom        float   %6.1f                 Headroom (in.)
trunk           int     %8.0g                 Trunk space (cu. ft.)
weight          int     %8.0gc                Weight (lbs.)
length          int     %8.0g                 Length (in.)
turn            int     %8.0g                 Turn Circle (ft.)
displacement    int     %8.0g                 Displacement (cu. in.)
gear_ratio      float   %6.2f                 Gear Ratio
foreign         byte    %8.0g       origin    Car type
mpg2            float   %9.0g       lmpg2     Mileage 2
-------------------------------------------------------------------------------
Sorted by: foreign
     Note: Dataset has changed since last saved.
```

Napominjemo da ove dvije komande lebel define i lebel values uvijek moraju pratiti jedna drugu, jer kreiranje oznaka bez pridruživanja istih vrijednostima varijable koju označavamo neće učiniti da se oznake prikazuju u rezultatima komandi kao što je npr. tab. Komandu tab ćemo detaljnije upoznati u nastavku Priručnika.

6. Komanda za opisivanje sadržaja baze podataka

6.1 Komanda describe

6.1.1 Opis

Komanda **describe** proizvodi sažeti prikaz baze podataka učitanih u memoriju ili podataka pohranjenih u specificiranoj datoteci u formatu Stata. Već smo ranije kroz Priručnik koristili ovu komandu, a ovdje samo navodimo njenu generičku sintaksu i dodatne opcije koje možete koristiti.

6.1.2 Sintaksa

Za opis podataka u memoriji:

describe [lista varijabli], [dodatne opcije memorije]

Za opis podataka iz datoteke Stata koja nije trenutno učitana u memoriju:

describe [lista varijabli] using nazivdatoteke, [dodatne opcije datoteke]

Dodatne opcije memorije su sljedeće:

Opcija	Opis
simple	prikazuje samo imena varijabli
short	prikazuje samo opšte informacije
fullnames	zabranjuje skraćivanje naziva varijabli
numbers	prikazuje broj varijable zajedno s imenom
replace	kreira bazu podataka, a ne pisani izvještaj, opis baze podataka
clear	za korištenje samo s opcijom replace
varlist	pohranjuje r(varlist) i r(sortlist) pored uobičajenih pohranjenih rezultata; opcije za programere

Dodatne opcije datoteke su sljedeće:

Opcija	Opis
short	prikazuje samo opšte informacije
simple	prikazuje samo imena varijabli
varlist	pohranjuje r(varlist) i r(sortlist) pored uobičajenih pohranjenih rezultata; opcija za programere

6.1.3 Primjer

Koristimo bazu podataka auto.dta. Nakon učitavanja baze podataka ukucamo komandu bez korištenja bilo kakvih opcija:

```
. describe

Contains data from C:\Program Files\Stata17\ado\base/a/auto.dta
 Observations:            74                  1978 automobile data
    Variables:            12                  13 Apr 2020 17:45
                                              (_dta has notes)
```

Variable name	Storage type	Display format	Value label	Variable label
make	str18	%-18s		Make and model
price	int	%8.0gc		Price
mpg	int	%8.0g		Mileage (mpg)
rep78	int	%8.0g		Repair record 1978
headroom	float	%6.1f		Headroom (in.)
trunk	int	%8.0g		Trunk space (cu. ft.)
weight	int	%8.0gc		Weight (lbs.)
length	int	%8.0g		Length (in.)
turn	int	%8.0g		Turn circle (ft.)
displacement	int	%8.0g		Displacement (cu. in.)
gear_ratio	float	%6.2f		Gear ratio
foreign	byte	%8.0g	origin	Car origin

```
Sorted by: foreign
```

Kao što vidimo u ispisu rezultata, ukupan broj opservacija u bazi je 74, te imamo ukupno 12 varijabli. Opis baze podataka nalazi se u gornjem desnom uglu ove tabele. U tabeli se potom navode naziv varijable, tip podataka (str znači string ili tekstualni tip podataka, int označava integer ili brojčani tip podataka izražen u cijelim brojevima, float označava tip podataka izražen u decimalnim brojevima, dok byte označava binarne varijable izražene u kategorijama 0 i 1). Potom se u tabeli prikazuje format u kojem se prikazuju podaci, kao i to da li je varijabli pridružena oznaka opservacija (kao npr. varijabli foreign gdje vidimo da je pridružena oznaka origin), te oznaka varijable koja se ispisuje u rezultatima. Ovdje vidimo jasnu razliku između naziva varijable i oznake varijable, gdje se u komandama koriste nazivi varijabli, a rezultati se ispisuju s oznakama varijabli, izuzev u slučajevima gdje to ne želimo što moramo dodatno naglasiti u samoj komandi koristeći opciju nolabel. Također u ispisu vidimo i da su podaci u bazi sortirani prema varijabli foreign.

Ukoliko su nazivi varijabli dugi, u prikazu će se pojaviti samo prva slova, ako želimo vidjeti puni naziv varijabli komandu je potrebno napisati u obliku: .describe, fullnames.

6.2 Komanda summarize

6.2.1 Opis

Komanda **summarize** izračunava i prikazuje razne mjere deskriptivne statistike. Ako nije navedena lista varijabli, statistika se izračunava za sve varijable u učitanoj bazi podataka.

6.2.2 Sintaksa

Generička sintaksa komande glasi:

summarize [lista varijabli] [if] [in] [weight], [opcije]

Dodatne opcije su sljedeće:

Opcija	Opis
detail	prikazuje dodatne statističke mjere
meanonly	zamjenjuje standardni prikaz; izračunava samo srednju vrijednost; opcija za programere
format	koristiti format prikaza varijable
separator(#)	nacrtati liniju razdvajanja nakon svake # varijable; zadani je separator(5)

6.2.3 Primjer

Učitamo bazu auto.dta te upišemo sljedeću komandu:

. summarize mpg

```
. summarize mpg
```

Variable	Obs	Mean	Std. dev.	Min	Max
mpg	74	21.2973	5.785503	12	41

Iz ispisa vidimo da smo dobili podatke o ukupnom broju opservacija koje imamo u varijabli mpg (74), srednjoj vrijednosti (21,2973), standardnoj devijaciji (5,785503), minimalnoj vrijednosti (12) i maksimalnoj vrijednosti (41).

Ako dodamo opciju detail, ispis će biti sljedeći:

```
. summarize mpg, detail
```

Mileage (mpg)

	Percentiles	Smallest		
1%	12	12		
5%	14	12		
10%	14	14	Obs	74
25%	18	14	Sum of wgt.	74
50%	20		Mean	21.2973
		Largest	Std. dev.	5.785503
75%	25	34		
90%	29	35	Variance	33.47205
95%	34	35	Skewness	.9487176
99%	41	41	Kurtosis	3.975005

Iz ispisa vidimo da smo dobili još dodatnih podataka o mjerama centralne tendencije (srednja vrijednost, percentili, kvartili, medijana), potom dodatne informacije o mjerama disperzije (varijansa i standardna devijacija), te mjeru asimetrije (Skewness) i mjeru spljoštenosti distribucije (Kurtosis).

Korištenje opcije separator dajemo u narednom primjeru:

```
. summarize, separator(4)
```

Variable	Obs	Mean	Std. dev.	Min	Max
make	0				
price	74	6165.257	2949.496	3291	15906
mpg	74	21.2973	5.785503	12	41
rep78	69	3.405797	.9899323	1	5
headroom	74	2.993243	.8459948	1.5	5
trunk	74	13.75676	4.277404	5	23
weight	74	3019.459	777.1936	1760	4840
length	74	187.9324	22.26634	142	233
turn	74	39.64865	4.399354	31	51
displacement	74	197.2973	91.83722	79	425
gear_ratio	74	3.014865	.4562871	2.19	3.89
foreign	74	.2972973	.4601885	0	1

U komandi nismo naveli listu varijabli pa smo dobili pregled svih varijabli u bazi. Također smo napravili razmak nakon svake četvrte varijable, te korištenjem ovakvih opcija možete pripremiti svoje tabele za direktno kopiranje u izvještaj. Iz ovog pregleda vidimo da kod svih varijabli imamo opservacije za sve modele, izuzev kod varijable rep78 gdje nam nedostaju podaci za 2 modela automobila. Također, vidimo i da za varijablu make nemamo opservacija, to je iz razloga što je ova varijabla tekstualnog oblika (tj. string), te se za nju ne mogu napraviti izračuni. Ukoliko imate podatke koji su u tekstualnom obliku s oznakama u bazi podataka, ovdje napominjemo da je iste potrebno pretvoriti u numerički oblik kako biste radili analize nad tim podacima u programu Stata. Ograničen je broj analiza koje je moguće uraditi u programu Stata korištenjem tekstualnih podataka.

6.3 Komanda tabulate

6.3.1 Opis

Komanda **tabulate** proizvodi tabelu jednostavnih ili unakrsnih rasporeda frekvencija, u ovisnosti od toga kako se koristi. Kratki naziv je tab.

6.3.2 Sintaksa

Ako se koristi za proizvodnju jednostavnih tabela rasporeda frekvencija, sintaksa je:

```
tabulate nazivvarijable
```

Ako se koristi za proizvodnju unakrsnih tabela rasporeda frekvencija, sintaksa je:

```
tabulate nazivvarijable1 nazivvarijable2
```

6.3.3 Primjer

Ovdje ćemo prvo prikazat jedan primjer gdje ćemo uporediti dvije komande: summarize i tabulate.

Kada koristite komandu summarize dobijate sljedeće informacije:

```
. summarize foreign
```

Variable	Obs	Mean	Std. Dev.	Min	Max
foreign	74	.2972973	.4601885	0	1

Iz ove tabele vidite koliko ukupno imamo opservacija, koja je aritmetička sredina, standardna devijacija, te minimalna i maksimalna vrijednost, ali ne znate koliko imamo domaćih a koliko stranih automobila u ovoj bazi podataka. To možete saznati koristeći komandu tabulate.

. tab foreign

```
. tab foreign
```

Car type	Freq.	Percent	Cum.
Domestic	52	70.27	70.27
Foreign	22	29.73	100.00
Total	74	100.00	

U ispisu vidimo da imamo 52 domaća i 22 strana automobila, procenat učešća u ukupnom broju podataka, kao i kumulativni procenat.

Korištenjem komande tab za kreiranje unakrsnih tabela frekvencija moguće je vidjeti npr. koliko imamo domaćih, a koliko stranih automobila po broju popravaka u 1978. godini:

. tab rep78 foreign

```
. tab rep78 foreign
```

Repair record 1978	Car origin		
	Domestic	Foreign	Total
1	2	0	2
2	8	0	8
3	27	3	30
4	9	9	18
5	2	9	11
Total	48	21	69

U ispisu vidimo da su po jedan popravak imala 2 automobila domaće proizvodnje, a da automobili strane proizvodnje nisu imali nijedan. Također imamo ispis ukupnog broja popravaka.

6.4 Komanda tabstat

6.4.1 Opis

Komanda **tabstat** prikazuje pregled izabranih statističkih mjera za seriju numeričkih varijabli u jednoj tabeli. Omogućava da se specificira lista statističkih pokazatelja koja će biti prikazana. Statistički pokazatelji mogu biti izračunati (uslovljeni) na osnovu druge varijable. Komanda tabstat omogućava određenu fleksibilnost u načinu i formatu tabele prikaza statističkih pokazatelja.

6.4.2 Sintaksa

tabstat nazivvarijable1,statistics (mjera) by (nazivvarijable2)

Korisne opcije koje se mogu specificirati su:

- by (nazivvarijable2) specificira koji statistički pokazatelji će biti prikazani odvojeno za svaku unikatnu vrijednost varijable gdje nazivvarijable2 može biti varijabla izražena u numeričkoj ili tekstualnoj vrijednosti.

- statistics(mjera) određuje koji statistički pokazatelji će biti prikazani; zadana postavka ispisuje aritmetičku sredinu. Pokazatelji u zagradi mogu biti određeni u zagradi listom pokazatelja koji su odvojeni praznim prostorom između. Neke od mogućnosti navodimo u tabeli:

Mjera	Opis
mean	Aritmetička sredina
count	Brojanje opservacija koje nisu nedostajuće
sum	Suma
max	Maksimalna vrijednost
min	Minimalna vrijednost
sd	Standardna devijacija

6.4.3 Primjer

I dalje koristimo bazu podataka auto.dta. Unutar baze imamo podatke o cijeni, težini, potrošnji goriva, i zapis o popravkama za 22 strana i 52 domaća automobila. Želimo izračunati prosječne vrijednosti ovih varijabli po porijeklu automobila. Unosimo sintaksu koja glasi:

.tabstat price weight mpg rep78, by(foreign)

Dobijamo sljedeći ispis:

```
. tabstat price weight mpg rep78, by( foreign)

Summary statistics: mean
  by categories of: foreign (Car type)
```

foreign	price	weight	mpg	rep78
Domestic	6072.423	3317.115	19.82692	3.020833
Foreign	6384.682	2315.909	24.77273	4.285714
Total	6165.257	3019.459	21.2973	3.405797

S obzirom na to da to nismo naglasili, u ovom slučaju u ispisu imamo prosječne vrijednosti za navedene varijable (price, weight, mpg, rep78) po njihovom porijeklu, kao i ukupni prosjek za sve automobile.

Više statističkih pokazatelja može biti prikazano korištenjem opcije statistics. Evo kako to izgleda na primjeru sintakse:

```
.tabstat price weight mpg rep78, by(foreign) stat(mean sd min max)
```

Dobijamo sljedeći ispis:

```
. tabstat price weight mpg rep78, by( foreign) stat(mean sd min max)

Summary statistics: mean, sd, min, max
  by categories of: foreign (Car type)
```

foreign	price	weight	mpg	rep78
Domestic	6072.423	3317.115	19.82692	3.020833
	3097.104	695.3637	4.743297	.837666
	3291	1800	12	1
	15906	4840	34	5
Foreign	6384.682	2315.909	24.77273	4.285714
	2621.915	433.0035	6.611187	.7171372
	3748	1760	14	3
	12990	3420	41	5
Total	6165.257	3019.459	21.2973	3.405797
	2949.496	777.1936	5.785503	.9899323
	3291	1760	12	1
	15906	4840	41	5

U ovom ispisu su nam za kategorije domaćih i stranih automobila redoslijedom prikazane aritmetička sredina, standardna devijacija, minimalna i maksimalna vrijednost. Također smo dobili isti ispis i za sve automobile u bazi u redu Total.

7. Komande za kreiranje grafikona

Prije prelaska na objašnjavanje pojedinačnih komandi za kreiranje grafikona, naglašavamo da se kreirani grafikoni iscrtavaju u odvojenom prozoru, te da je nakon korištenja određene komande svaki potrebno spasiti u formatu u kojem želimo, a prije pokretanja komande za crtanje narednog grafikona bez obzira na to da li koristimo iste ili druge varijable.

7.1 Komanda graph bar i graph hbar

7.1.1 Opis

Komanda **graph bar** je komanda koja iscrtava vertikalni dijagram sa stupcima. Na vertikalnom dijagramu sa stupcima, y osa je numerička, a x osa je kategorijska varijabla.

Komanda **graph hbar** je komanda za iscrtavanje horizontalnog dijagrama sa stupcima. Na horizontalnom dijagramu sa stupcima, numerička osa je i dalje y osa, a kategorijska osa je i dalje x osa, s tim da se y osa prikazuje horizontalno, a x osa vertikalno.

7.1.2 Sintaksa

Osnovna sintaksa glasi ovako:

```
graph bar yvars [if]  [in]  [weight]  [,options]
```

U primjerima navodimo nekoliko korisnih opcija koje se koriste za samo uređivanje izgleda grafikona. Za sve primjere i dalje koristimo bazu auto.dta koja dolazi predinstalirana sa softverom Stata.

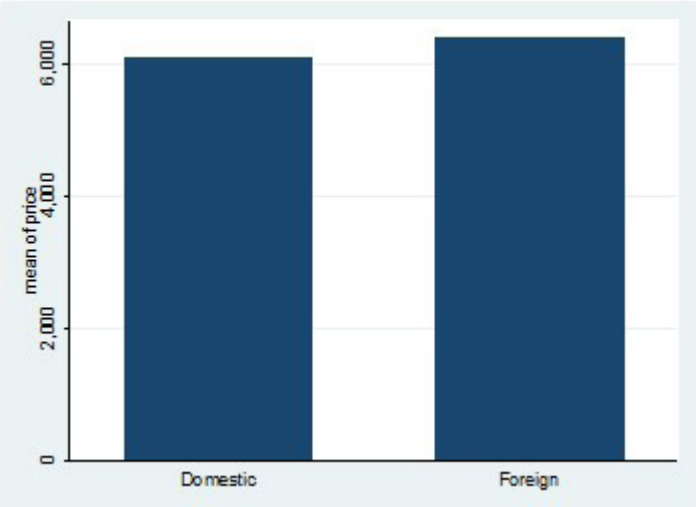
7.1.3 Primjer

Učitamo bazu podataka auto.dta. Želimo prezentirati grafički podatke o cijeni i težini za domaće i inostrane automobile u bazi. Dakle radimo s tri različite varijable: price i weight (numeričke varijable) i foreign (kategorijska varijabla). Pokazat ćemo nekoliko opcija za kreiranje vertikalnog dijagrama sa stupcima, a sve opcije se mogu primijeniti i za horizontalni dijagram sa stupcima (dat ćemo jedan prikaz i ovog grafikona).

Za početak ćemo kreirati grafikon Prosječna cijena automobila u odnosu na njegovo porijeklo. Sintaksa je sljedeća:

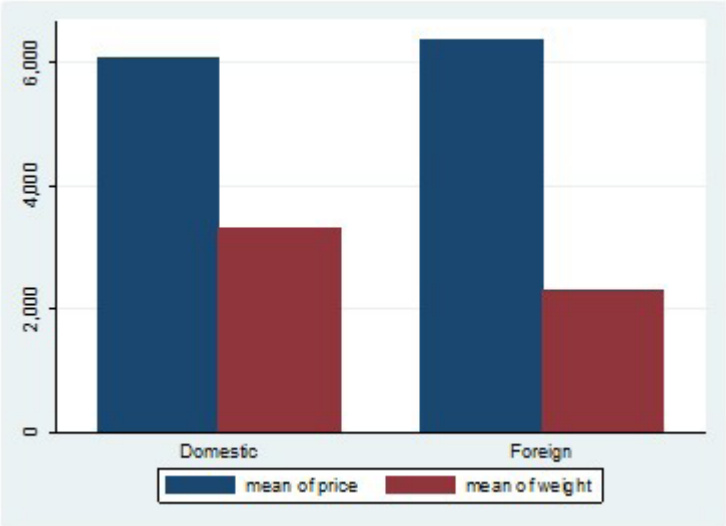
```
. graph bar (mean) price, over(foreign)
```

Dobijamo sljedeći grafikon:



Na sljedećem grafikonu dodat ćemo još jednu varijablu koja se odnosi na prosječnu težinu automobila u zavisnosti od porijekla automobila:

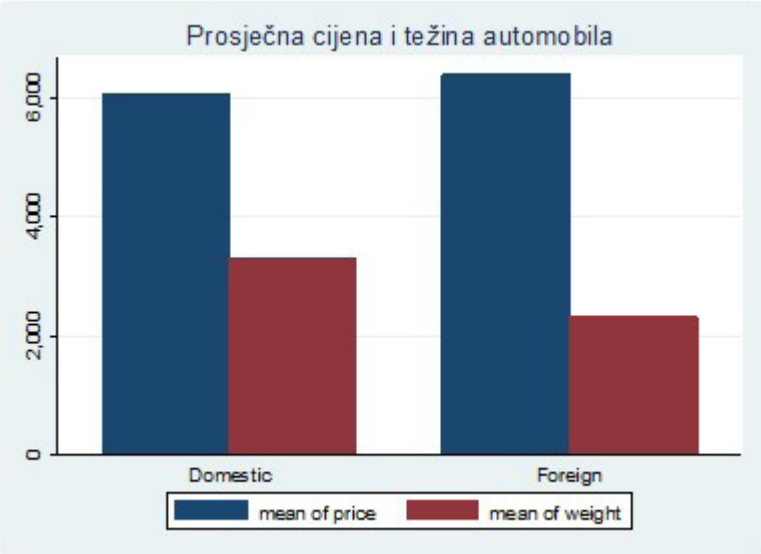
```
. graph bar (mean) price weight, over(foreign)
```



Kako bi ovi grafikoni izgledali onako kako ih često susrećemo u literaturi, potrebno je iskoristiti opcije za uređivanje izgleda grafikona. Pokazat ćemo samo neke osnovne koje se često upotrebljavaju, a postoji niz opcija koje je moguće iskoristiti.

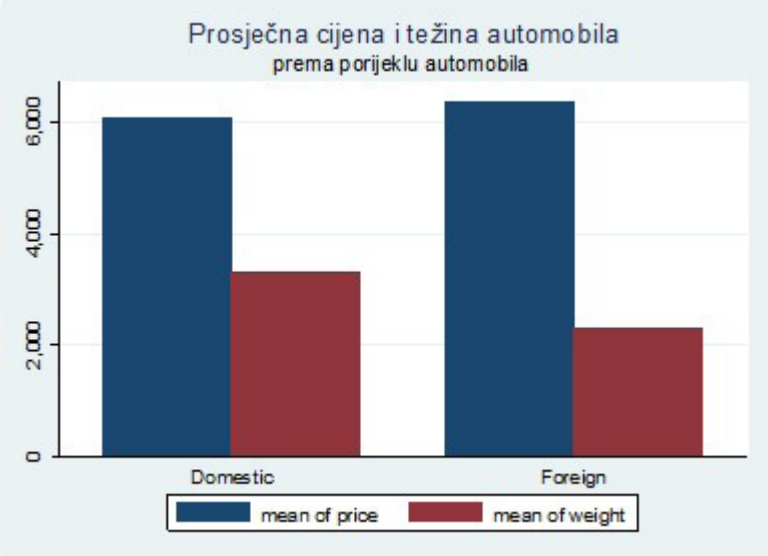
Postavljanje naziva grafikona moguće je korištenjem opcije title(„text of title“). Uzimamo prethodni grafikon i dodajemo naziv „Prosječna cijena i težina automobila“.

. graph bar (mean) price weight, over (foreign) title (“Prosječna cijena i težina automobila”)



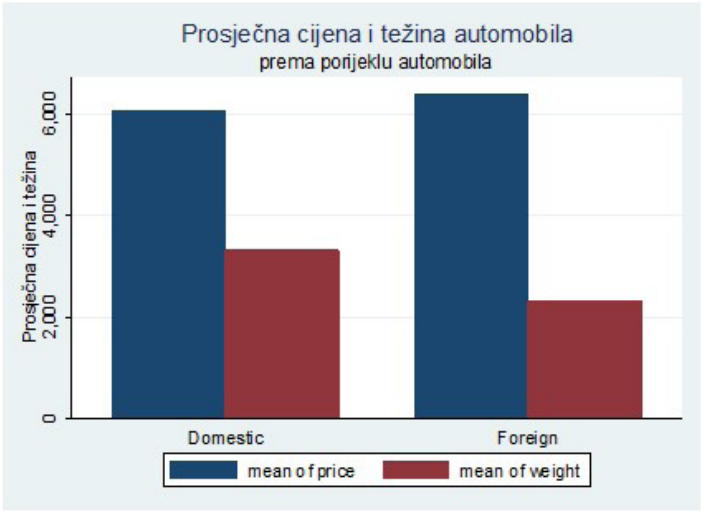
Dodat ćemo podnaslov na grafikon „prema porijeklu automobila“ korištenjem opcije subtitle(„text of subtitle“).

. graph bar (mean) price weight, over (foreign) title (“Prosječna cijena i težina automobila”) subtitle (“prema porijeklu automobila”)



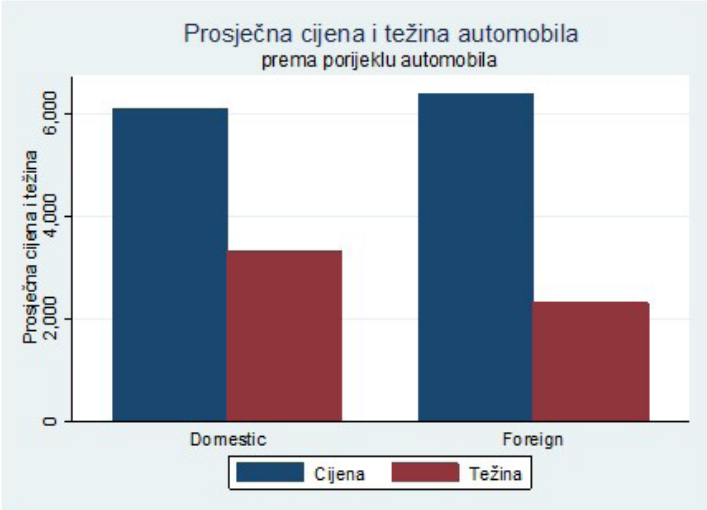
Za dodavanje naziva y-ose koristimo opciju ytitle („text of ytitle“). Dodat ćemo naziv y-ose „Prosječna cijena i težina“.

```
. graph bar (mean) price weight, over (foreign) title (“Prosječna cijena i težina automobila”) subtitle (“prema porijeklu automobila”) ytitle (“Prosječna cijena i težina”)
```



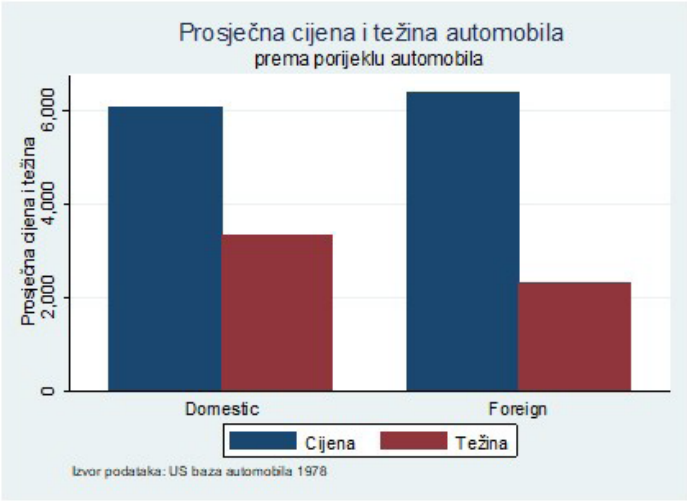
Za uređivanje legende koristimo opciju legend(label()).

```
.graph bar (mean) price weight, over (foreign) title (“Prosječna cijena i težina automobila”) subtitle (“prema porijeklu automobila”) ytitle (“Prosječna cijena i težina”) legend ( label(1 “Cijena”) label (2 “Težina”) )
```



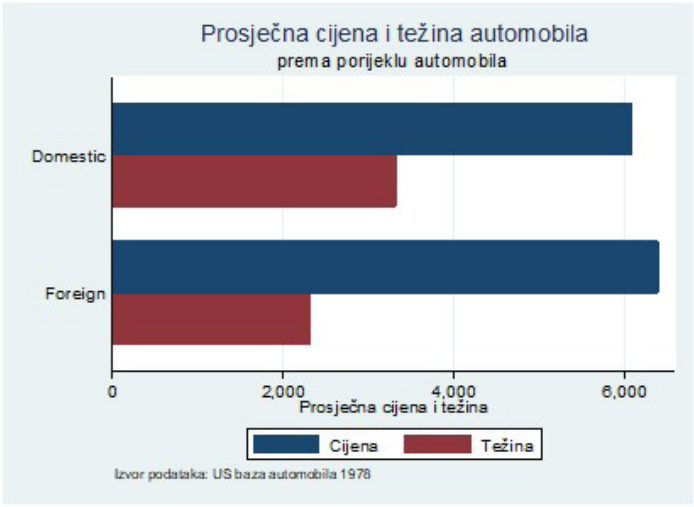
I na samom kraju dodat ćemo još i Izvor podataka ispod grafikona koristeći opciju note(„text note“).

.graph bar (mean) price weight, over (foreign) title (“Prosječna cijena i težina automobila”) subtitle (“prema porijeklu automobila”) ytitle (“Prosječna cijena i težina”) legend (label(1 “Cijena”) label (2 “Težina”) note (“Izvor podataka: US baza automobila 1978”)



Iskoristit ćemo posljednju sintaksu da pokažemo i kako izgleda horizontalni dijagram sa stupcima.

.graph hbar (mean) price weight, over (foreign) title (“Prosječna cijena i težina automobila”) subtitle (“prema porijeklu automobila”) ytitle (“Prosječna cijena i težina”) legend (label(1 “Cijena”) label(2 “Težina”)) note (“Izvor podataka: US baza automobila 1978”)



7.2 Komanda graph pie

7.2.1 Opis

Komanda **graph pie** iscrtava „pita“ grafikone.
Ova komanda ima tri načina izrade.

Prvi način je da specificiramo dvije ili više različitih varijabli: graph pie var1 var2 var3. Na ovaj način se iscrtaju tri grafikona, prvi koji odgovara sumi var1, drugi sumi var2, a treći sumi var3.

Drugi način je kada se specificira jedna varijabla s opcijom over(): graph pie var1, over(var2). Dijelovi „pita“ grafikona iscrtani su za svaku vrijednost varijable var2. Prvi dio pite odgovara sumi var1 za prvu grupu varijable var2; drugi dio pite odgovara sumi var1 za drugu grupu varijable var2, itd.

Treći način je specifikacija opcije over() ali bez navođenja varijabli: graph pie, over(var1). Dijelovi pite se iscrtavaju za svaku vrijednost var1, a kriške odgovaraju broju opservacija svake grupe.

7.2.2 Sintaksa

S obzirom na to da postoje tri načina, navodimo osnovni oblik sintakse za svaki način:

Dijelovi pite kao sume ili procenti svake varijable

graph pie varlist [if] [in] [weight] [,options]

Dijelovi pite kao sume ili procenti kategorija korištenjem opcije over()

graph pie varname [if] [in] [weight],over(varname) [options]

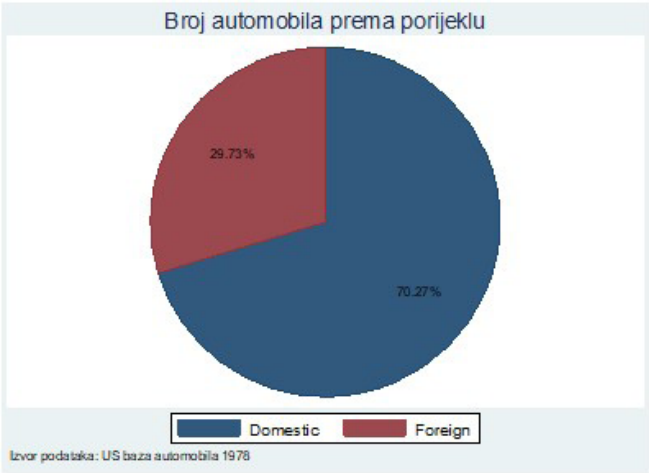
Dijelovi pite kao frekvencije korištenjem opcije over ()

graph pie [if] [in] [weight],over(varname) [options]

7.2.3 Primjer

Koristimo bazu podataka auto.dta. Na „pita“ grafikonu želimo prikazati procenat automobila prema njihovom porijeklu. Koristit ćemo treći način izrade grafikona.

.graph pie, over (foreign) title (“Broj automobila prema porijeklu”) plabel (_all percent) note (“Izvor podataka: US baza automobila 1978”)



Opisane opcije za crtanje dijagrama sa stupcima mogu se koristiti i prilikom crtanja „pita“ grafikona, osim opcije za ytitle i legend, te opcije legend (off).

Uz opciju plabel mogu se iskoristiti sljedeće opcije: format (_all sum ili all_percent), gap, textbox_options.

7.3 Komanda graph twoway

7.3.1 Opis

Komanda **graph twoway j**e komanda koja se koristi za iscrtavanje porodice grafikona gdje su i x i y varijable numeričke vrijednosti.

S obzirom na to da postoji lista različitih grafikona koji se mogu kreirati korištenjem ove komande, zasada ćemo izdvojiti sljedeće: dijagram rasipanja (scatter plot), poligon frekvencija, dijagram povezanih tačaka, histogram.

7.3.2 Sintaksa

Osnovna sintaksa za sve grafikone iz ove porodice grafikona je:

[graph] twoway plot [if] [in] [,twoway_options]

gdje je plot tip grafikona koji se kreira, npr. scatter (dijagram rasipanja), line (poligon frekvencija), connected (dijagram povezanih tačaka), histogram (histogram).

Uz sve navedene tipove moguće je dodati niz opcija za uređivanje naziva osa, naziva grafikona, legende i slične opcije.

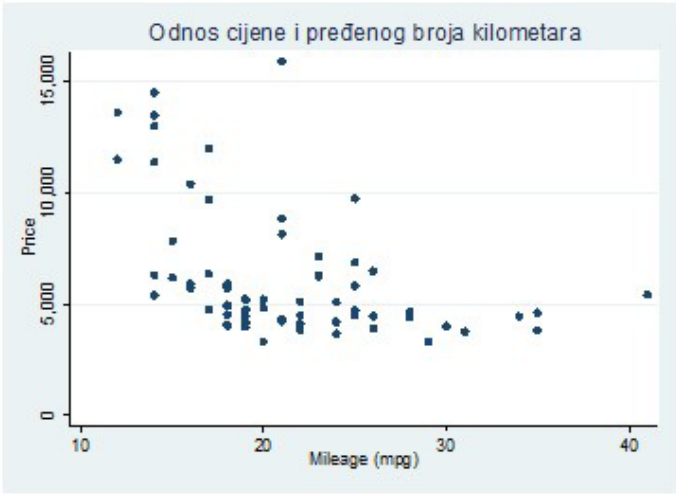
7.3.3 Primjer – scatter

Sintaksa za iscrtavanje ovog grafikona je:

[twoway] scatter varlist [if] [in][weight] [,options] gdje je varlist lista varijabli u sljedećem redoslijedu y1, y2... x.

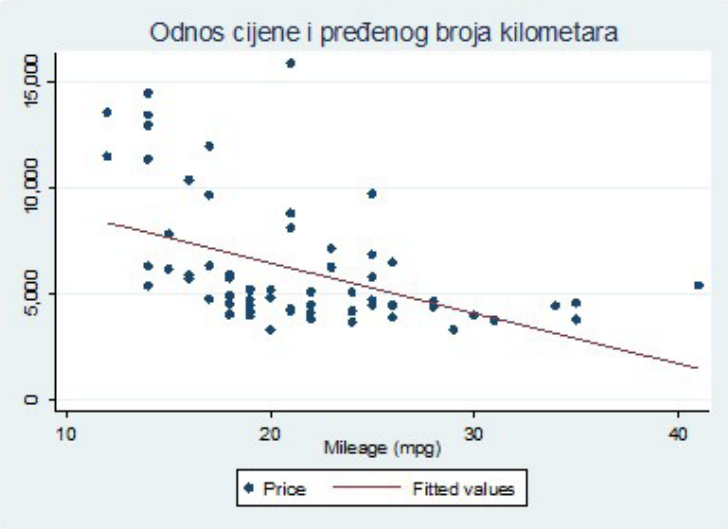
Koristit ćemo bazu podataka auto.dta. Želimo nacrtati dijagram rasipanja za varijable price (y varijabla) i mpg (x varijabla).

. twoway scatter price mpg, title("Odnos cijene i pređenog broja kilometara")



Moguće je dodati i ocijenjenu regresionu pravu na ovaj grafikon i to radimo na sljedeći način:

.twoway scatter price mpg || lfit price mpg, title("Odnos cijene i pređenog broja kilometara")



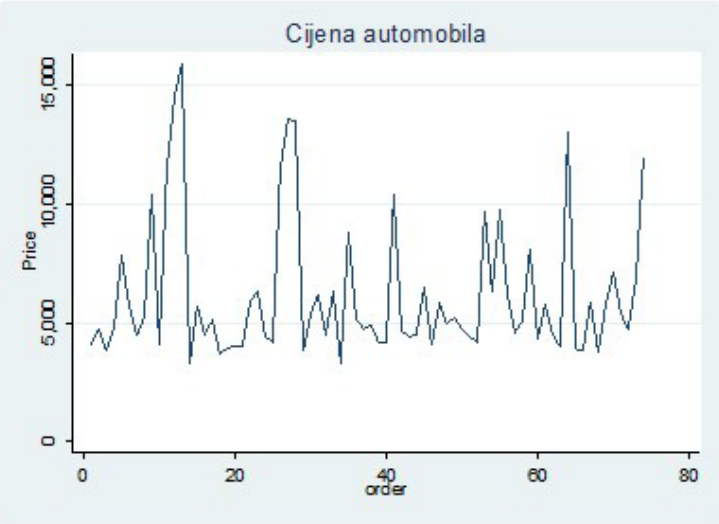
7.3.4 Primjer - line

Sintaksa za iscrtavanje ovog grafikona je:

[twoway] line varlist [if] [in] [,options] gdje je varlist lista varijabli u sljedećem redoslijedu y1, y2... x.

Koristimo bazu auto.dta. Želimo nacrtati poligon frekvencija, koji se često koristi i za iscrtavanje vremenskih linija. S obzirom na to da u ovoj bazi nemamo odgovarajućih varijabli da bi dobili ovakav grafikon, možemo kreirati varijablu pod nazivom order gdje smo poredali sve automobile u bazi po nazivu. Potom smo iscrtali grafikon varijable price u odnosu na varijablu order.

```
. gen order=_n
. twoway line price order, title("Cijena automobila")
```



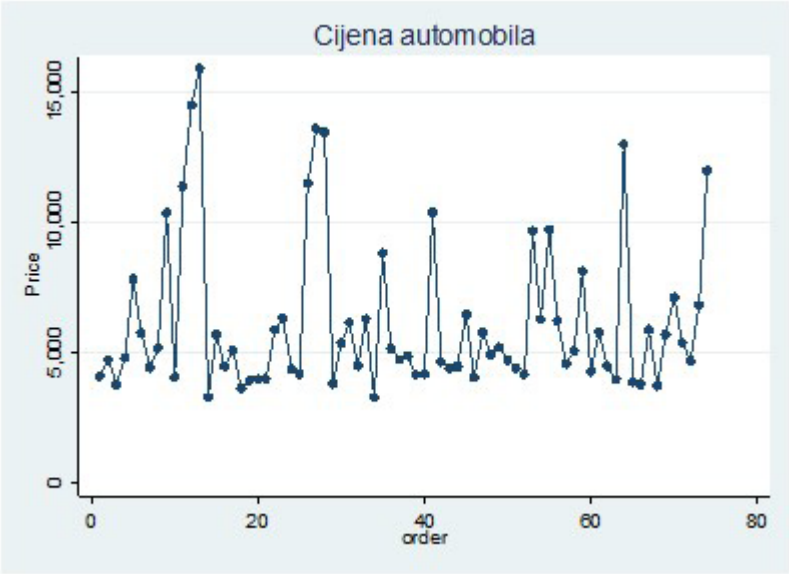
7.3.5 Primjer - connected

Sintaksa za ovaj grafikon je:

[twoway] connected varlist [if] [in] [weight] [,scatter_options] gdje je varlist lista varijabli u sljedećem redoslijedu y1, y2... x.

Koristimo bazu auto.dta. Želimo nacrtati dijagram povezanih tačaka. Koristimo varijablu order koju smo kreirali u prethodnom primjeru i crtamo grafikon varijable price u odnosu na varijablu order.

```
. twoway connected price order, title("Cijena automobila")
```



7.3.6 Primjer - histogram

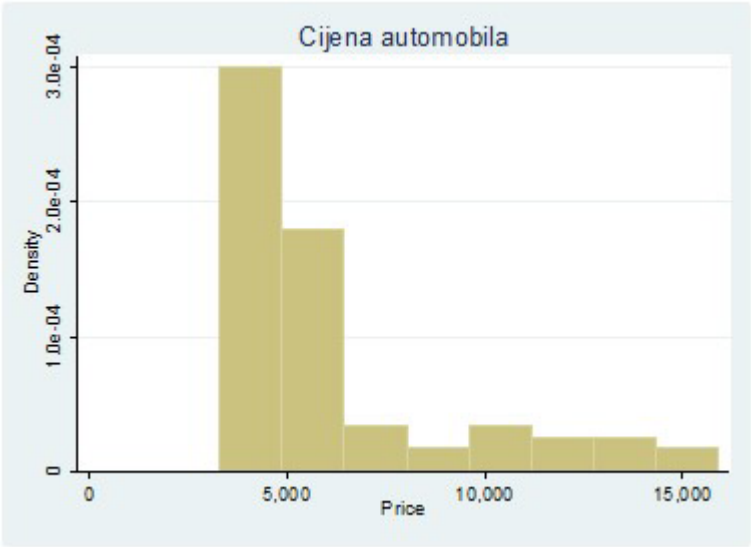
Sintaksa ove komande glasi:

```
histogram nazivvarijable [if] [in] [weight] [,options]
```

Ako u komandi nije specificirano da li je riječ o kontinuiranoj ili diskretnoj varijabli, histogram će iscrtati varijablu kao da je riječ o kontinuiranoj varijabli.

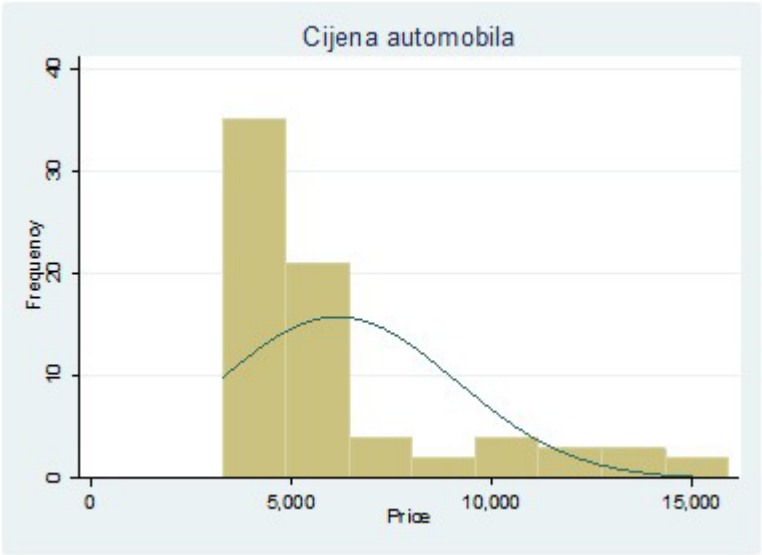
Uzet ćemo ponovo primjer iz baze auto.dta. Crtamo histogram za varijablu price.

```
. histogram price, title("Cijena automobila")
```



Za dodavanje krive normalne distribucije uradimo sljedeće:

. histogram price, freq normal title(“Cijena automobila”)



8. Komande za korelacionu analizu

Korelaciona analiza je statistička metoda koja se koristi za mjerenje jačine linearne veze između dvije varijable i izračunavanje njihove povezanosti. Jednostavno rečeno – korelacionom analizom se izračunava nivo promjene jedne varijable uslovljen promjenom u drugoj varijabli. Visoka korelacija ukazuje na jaku vezu između dvije varijable, dok niska korelacija znači da su varijable slabo povezane. Obično se mjeri različitim koeficijentima, od čega je najpoznatiji Pearsonov korelacioni koeficijent.

Postoje različiti način za utvrđivanje korelacija između varijabli, počevši od iscrtavanja grafikona koji pokazuje odnos između dvije varijable (npr. dijagram rasipanja – scatter), te korištenjem komandi poput correlate i pwcorr koje objašnjavamo u nastavku.

8.1 Opis

Komanda **correlate** prikazuje matricu korelacije ili matricu kovarijanse za grupu varijabli. Ako grupa varijabli nije specificirana, matrica se prikazuje za sve varijable u bazi podataka.

Komanda **pwcorr** prikazuje sve koeficijente parne korelacije između varijabli u navedenoj listi varijabli ili, ako lista nije specificirana, za sve varijable u učitanoj bazi podataka.

8.2 Sintaksa

Za prikaz korelacione matrice ili matrice kovarijanse:

correlate [lista_varijabli] [if] [in] [weight], [correlate_opcije]

Sljedeće opcije mogu se koristiti uz komandu correlate:

Opcija	Opis
means	prikazuje aritmetičku sredinu, standardnu devijaciju, minimum i maksimum zajedno s matricom
noformat	zanemaruje format prikaza povezan s varijablama
covariance	prikazuje kovarijansu
wrap	dozvoljava da se široke matrice razlome

Za prikaz svih koeficijenata parne korelacije:

pwcorr [lista_varijabli] [if] [in] [weight], [pwcorr_opcije]

Sljedeće opcije mogu se koristiti uz komandu pwcorr:

Opcija	Opis
obs	ispisuje broj opservacija za svaki unos
sig	ispisuje nivo značajnosti za svaki unos
listwise	koristite opciju za brisanje vrijednosti koje nedostaju
casewise	sinonim za listwise
print(#)	nivoi značajnosti za koeficijente koji se ispisuju
star(#)	nivo značajnosti koji će biti označen sa zvjezdicom
bonferroni	korištenje Bonferroni prilagođenog nivoa značajnosti
sidak	korištenje Šid´ak prilagođenog nivoa značajnosti

8.3 Primjer

Ukoliko želimo da koristimo komandu correlate bez navođenja listi varijabli, onda dobijamo sljedeći rezultat u ispisu iz naše baze podataka auto.dta:

```
. correlate
(make ignored because string variable)
(obs=69)
```

	price	mpg	rep78	headroom	trunk	weight	length	turn	displa~t	gear_r~o	foreign
price	1.0000										
mpg	-0.4559	1.0000									
rep78	0.0066	0.4023	1.0000								
headroom	0.1112	-0.3996	-0.1480	1.0000							
trunk	0.3232	-0.5798	-0.1572	0.6608	1.0000						
weight	0.5478	-0.8055	-0.4003	0.4795	0.6691	1.0000					
length	0.4425	-0.8037	-0.3606	0.5240	0.7326	0.9478	1.0000				
turn	0.3302	-0.7355	-0.4961	0.4347	0.6008	0.8610	0.8631	1.0000			
displacement	0.5479	-0.7434	-0.4119	0.4763	0.6287	0.9316	0.8621	0.8124	1.0000		
gear_ratio	-0.3802	0.6565	0.4103	-0.3790	-0.5107	-0.7906	-0.7232	-0.7005	-0.8381	1.0000	
foreign	-0.0174	0.4538	0.5922	-0.3347	-0.4053	-0.6460	-0.6110	-0.6768	-0.6383	0.7266	1.0000

Kao što vidimo, varijabla make je ignorisana jer sadrži isključivo tekstualne podatke. Ukupan broj opservacija za koji je izračunat koeficijent iznosi 69, što je manje od ukupnog broja opservacija u cjelokupnoj bazi (prisjetite se da je to 74). Ovo pokazuje da korištenjem komande correlate možemo izračunati koeficijente samo za varijable kod kojih imamo unijete sve vrijednosti. Ukoliko npr. navedemo listu varijabli koje sadržavaju sve vrijednosti (prisjetite se da samo varijabla rep78 ima 69 opservacija) dobijamo sljedeći ispis:

.correlate price mpg foreign

Dobijamo sljedeći ispis:

```
. correlate price mpg foreign
(obs=74)
```

	price	mpg	foreign
price	1.0000		
mpg	-0.4686	1.0000	
foreign	0.0487	0.3934	1.0000

Vidimo da za razliku od prethodnog ispisa, u izračunu sada učestvuju ukupno 74 opservacije, te su koeficijenti naravno nešto drugačiji u odnosu na prethodni ispis.

Komanda **correlate** izračunava koeficijente korelacije korištenjem procedure brisanja slučajeva; kada želite izračunati koeficijente korelacije varijabli x1, x2, : : : , xk, ako za bilo koju varijablu nedostaje neka od opservacija, bilo koja od njih se ne koristi. Dakle, ako x3 i x4 nemaju vrijednosti koje nedostaju, ali x2 nedostaje za polovicu podataka, korelacija između x3 i x4 se izračunava koristeći samo polovicu podataka za koje x2 ne nedostaje. Naravno, možete dobiti korelaciju između x3 i x4 koristeći sve podatke upisivanjem correlate x3 x4.

Komanda **pwcorr** olakšava dobijanje takvih koeficijenata primjenjujući princip parne korelacije.

Upišite sljedeću sintaksu:

.pwcorr mpg price rep78 foreign, obs sig

```
. pwcorr mpg price rep78 foreign, obs sig
```

	mpg	price	rep78	foreign
mpg	1.0000			
	74			
price	-0.4686	1.0000		
	0.0000			
	74	74		
rep78	0.4023	0.0066	1.0000	
	0.0006	0.9574		
	69	69	69	
foreign	0.3934	0.0487	0.5922	1.0000
	0.0005	0.6802	0.0000	
	74	74	69	74

Kao što vidimo, za razliku od prethodnog primjera koji koristi correlate, pwcorr koristi sve moguće opservacije koje su dostupne za izračun, za sve varijable gdje imamo puni broj opservacija, koristimo sve opservacije, dok kod onih koje nemaju puni broj slučajeva (npr. varijabla rep78) koristimo dostupni broj opservacija za izračun korelacionog koeficijenta. U prethodnom ispisu prikazan je i nivo značajnosti za navedene izračunate korelacione koeficijente.

U nastavku dajemo nešto širu prezentaciju rezultata s označavanjem nivoa značajnosti u odnosu na prethodno postavljenu granicu.

.pwcorr mpg price headroom trunk rep78 foreign, print(.05) star(.01)

```
. pwcorr mpg price headroom trunk rep78 foreign, print(.05) star(.01)

             |      mpg    price headroom    trunk    rep78  foreign
-------------+------------------------------------------------------
         mpg |   1.0000
       price |  -0.4686*  1.0000
    headroom |  -0.4138*           1.0000
       trunk |  -0.5816*  0.3143*  0.6620*  1.0000
       rep78 |   0.4023*                     1.0000
     foreign |   0.3934*          -0.2939  -0.3594*  0.5922*  1.0000
```

U navedenom primjeru zvjezdicom su označeni oni koeficijenti koji su značajni na nivou p=0,01.

9. Regresiona analiza

Regresiona analiza je statistička tehnika koja ispituje vezu između jedne neprekidne zavisne (y) i jedne ili više nezavisnih neprekidnih/prekidnih (x) varijabli. Osnovni cilj regresione analize je da testira jednu ili više različitih hipoteza i da se koristi za predviđanje.

Opšti oblik regresionog modela je:

$Y=f(X\_1,X\_2,\ldots,X\_K)+e$

gdje je Y zavisna varijabla, X su nezavisne varijable i parametar e je slučajno odstupanje. Jednostavni regresioni model ima sljedeći oblik:

$Y=f(X)+e.$

Veze između varijabli mogu biti različite, ali za potrebe ovog Priručnika ograničit ćemo se na prezentaciju jednostavnog linearnog regresionog modela ocijenjenog metodom najmanjih kvadranata (metod OLS), čija funkcionalna forma glasi:

$E(Y\_i)=a+bX\_i.$

Zadatak regresione analize je ocjena modela, dok je zadatak korelacione analize utvrđivanje stepena i smjera povezanosti pojava.

Kao rezultat korištenja komandi u programu Stata za ocjenu regresionog modela pojavljuje se i koeficijent determinacije R2. Vrijednost ovog koeficijenta se kreće između nule i jedinice. On pokazuje koja je proporcija ukupne varijacije varijable Y objašnjena ocijenjenom regresionom jednačinom i uobičajeno je da se izražava u procentima. Veća vrijednost ovog koeficijenta ukazuje da je veća proporcija objašnjena u ukupnoj varijaciji i da je odabrani model pouzdaniji i reprezentativniji.

S obzirom na to da regresiona analiza polazi od nekoliko pretpostavki koje se moraju ispuniti da bi model bio ispravan, iste je potrebno testirati kako bi ocijenili model. Pretpostavke su vezane za korištenje metoda najmanjih kvadrata i danas je ta teorema poznata pod nazivom Gauss-Markov teorem. Gauss-Markovi uslovi su da greška ima uslovnu aritmetičku sredinu jednaku nuli, da greška ima konstantnu varijansu i da greške ne koreliraju za različite opservacije i s X varijablama.

Pored uslova Gauss-Markovog teorema uobičajeno provjeravamo i da li je model ispravno specificiran, odsutnost multikolinearnosti i da li su reziduali normalno distribuirani.

Uz objašnjenje izabranih komandi objasnit ćemo i uslove koje one provjeravaju. Naglašavamo da komande za dijagnostičke testove moraju slijediti nakon komande za regresionu analizu.

9.1 Komanda regress

9.1.1 Opis

Komanda **regress** koristi se za ocjenu modela linearne regresije metodom najmanjih kvadranata. regress se može koristiti i za ponderisanu ocjenu, izračun robusnih i klaster-robusnih standardnih grešaka, te prilagođavanje rezultata za kompleksne dizajne istraživanja.

9.1.2 Sintaksa

Osnovna sintaksa za ocjenu linearnog regresionog modela glasi:

regress depvar [indepvars][if][in][weight][,options]

gdje depvar označava zavisnu varijablu, a indepvars označava listu nezavisnih varijabli, a sintaksa se može proširiti listom različitih opcija.

9.1.3 Primjer

Koristimo bazu podataka auto.dta. Želimo da uspostavimo funkcionalnu vezu između cijene kao zavisne varijable automobila i broja pređenih kilometara. Koristimo sljedeću komandu:

. regress price mpg

Rezultat ove komande izgleda ovako:

Source	SS	df	MS	Number of obs	=	74
				F(1, 72)	=	20.26
Model	139449474	1	139449474	Prob > F	=	0.0000
Residual	495615923	72	6883554.48	R-squared	=	0.2196
				Adj R-squared	=	0.2087
Total	635065396	73	8699525.97	Root MSE	=	2623.7

price	Coef.	Std. Err.	t	P>|t|	[95% Conf.	Interval]
mpg	-238.8943	53.07669	-4.50	0.000	-344.7008	-133.0879
_cons	11253.06	1170.813	9.61	0.000	8919.088	13587.03

Kao što vidimo, regress je komanda koja kreira 3 tabele s rezultatima. Prva tabela (gore lijevo) je tabela analize varijanse (ANOVA). Naslovi kolona su zapravo SS-suma kvadrata (sum of squares), df-stepeni slobode (degrees of freedom) i MS-aritmetička sredina kvadrata (mean square). U ovom primjeru ukupna suma kvadrata je 635065396, od čega je modelom objašnjeno 139449474, a 495615923 je ostavljeno neobjašnjeno. Zbog toga što regresija uključuje konstantu, ukupna suma reflektuje sumu nakon uklanjanja aritmetičkih sredina, kao i suma objašnjena modelom. Tabela također pokazuje da postoje 73 stepena slobode (koji se broje kada se od 74 opservacije oduzme 1 zbog uklanjanja sredina), od čega je 1 iskorištena za model, a 72 za rezidual.

Desno se nalazi tabela ANOVA koja prikazuje drugi pregled sumarne statistike. F statistika povezana s tabelom ANOVA je 20,26. Za stepene slobode možemo reći da je brojnik jednak 1, dok je nazivnik jednak 72. F statistika testira hipotezu da li su svi koeficijenti isključujući konstantu jednaki nuli. Vjerovatnoća da je F statistike velika ili veća označena je s 0,0000 što je način programa Stata da označi broj manji od 0,00005. R2 za ovu regresiju je 0,2196, a R2 prilagođen za stepene slobode je 0,2087. Korijen srednje kvadratne greške označen kao Root MSE je 2623,7. Ovo je kvadratni korijen srednje kvadratne greške za rezidual u tabeli ANOVA.

Konačno, Stata proizvodi tabelu ocijenjenih koeficijenata. Prvi red tabele pokazuje da je kao zavisna varijabla određena varijabla price. Potom se redaju ocijenjeni koeficijenti. Naš model bi prema tome bio:

$$price=11253,06-238,8943\cdot mpg$$

Desno od koeficijenata je ispis standardnih grešaka. Na primjer, standardna greška za koeficijent uz mpg je 53,07669. Odgovarajuća t statistika je -4,50, koja odgovara nivou dvostrane statističke značajnosti 0,000. Ovaj broj pokazuje da je značajnost manja od 0,0005. 95% interval povjerenja za ovaj koeficijent je [-344,7008, -133,0879].

9.2 Komanda linktest i estat ovtest

9.2.1 Opis

Obje komande koriste se da testiramo pretpostavku da li su sve relevantne X-varijable uključene u model.

Komanda **linktest** nam daje regresionu analizu s dvije varijable: _hat (linearno predviđena vrijednost) koja ukoliko imamo dobar model treba da bude dobar prediktor Y varijable; i _hatsq (kvadratno predviđena vrijednost) koja ne treba da ima statističku značajnost ako je model ispravno specificiran. Posljednja varijabla označava ovaj test na sljedeći način: ako je _hatsq statistički značajna, onda je i linktest značajan, što znači da smo

izostavili relevantne varijable i/ili da model nije korektno specificiran.

Komanda **estat** ovtest prikazuje dvije verzije Ramsey regresiono specificiranog testa grešaka za izostavljene varijable. Test koji nije statistički značajan znači da možemo zadržati nultu hipotezu da nema izostavljenih varijabli, tj. da je prema ovom testu naš model dobro specificiran.

9.2.2 Sintaksa

Oblik sintakse jednak je nazivu komande tj. linktest i estat ovtest.

9.2.3 Primjer

Želimo kreirati i ispitati regresioni model sljedećih varijabli: mpg, weight, displacement i foreign, te testirati model.

```
. regress mpg weight displacement foreign
. linktest
```

Source	SS	df	MS	Number of obs	=	74
				F(3, 70)	=	45.88
Model	1619.71935	3	539.906448	Prob > F	=	0.0000
Residual	823.740114	70	11.7677159	R-squared	=	0.6629
				Adj R-squared	=	0.6484
Total	2443.45946	73	33.4720474	Root MSE	=	3.4304

mpg	Coef.	Std. Err.	t	P>|t|	[95% Conf.	Interval]
weight	-.0067745	.0011665	-5.81	0.000	-.0091011	-.0044479
displacement	.0019286	.0100701	0.19	0.849	-.0181556	.0220129
foreign	-1.600631	1.113648	-1.44	0.155	-3.821732	.6204699
_cons	41.84795	2.350704	17.80	0.000	37.15962	46.53628

Source	SS	df	MS	Number of obs	=	74
				F(2, 71)	=	76.75
Model	1670.71514	2	835.357572	Prob > F	=	0.0000
Residual	772.744316	71	10.8837228	R-squared	=	0.6837
				Adj R-squared	=	0.6748
Total	2443.45946	73	33.4720474	Root MSE	=	3.299

mpg	Coef.	Std. Err.	t	P>|t|	[95% Conf.	Interval]
_hat	-.4127198	.6577736	-0.63	0.532	-1.724283	.8988434
_hatsq	.0338198	.015624	2.16	0.034	.0026664	.0649732
_cons	14.00705	6.713276	2.09	0.041	.6211539	27.39294

Što se tiče našeg primjera, vidimo da je linktest statistički značajan, što znači da smo ili izostavili relevantne varijable, ili nezavisne varijable nisu dobro specificirane.

Rezultati Ramsey testa u našem primjeru su kako slijedi:

```
. regress mpg weight displacement foreign

. estat ovtest
```

```
Ramsey RESET test using powers of the fitted values of mpg
       Ho:  model has no omitted variables
                  F(3, 67) =      2.53
                  Prob > F =      0.0643
```

Kao što možemo vidjeti, test nije statistički značajan, odnosno nema izostavljenih varijabli, pa je prema ovom testu naš model dobro specificiran.

9.3 Komanda estat vif i estat vce

9.3.1 Opis

Komande **estat vif** i **estat vce** koriste se za ispitivanje multikolinearnosti između varijabli u modelu, s obzirom na to da pretpostavka odsutnosti multikolinearnosti postoji ako dvije X-varijable u istom modelu nisu perfektno korelirane i ako jedna X-varijabla ne može biti perfektno objašnjenja linearnom kombinacijom drugih X-varijabli u našem modelu.

Komanda **estat vif** koristi se da bi dobili inflacijski faktor varijanse (VIF) i tolerantne vrijednosti. Ako je VIF vrijednost veća od 5 za bilo koju varijablu, možemo imati problem multikolinearnosti.

Za detaljniju analizu multikolinearnosti potrebno je koristiti **estat vce** za kreiranje matrice varijansi-kovarijansi.

9.3.2 Sintaksa

Oblik sintakse jednak je nazivu komande, tj. estat vif i estat vce. Pošto je zadana matrica kod komande estat vce matrica kovarijansi, da bi dobili matricu korelacionih koeficijenata u sintaksu je potrebno dodati sljedeće: estat vce, correlation.

9.3.3 Primjer

Nastavljamo s istim modelom kao u prethodnom primjeru i koristimo komande za testiranje modela.

. regress mpg weight displacement foreign

. estat vif

Rezultat je sljedeći:

Variable	VIF	1/VIF
displacement	5.31	0.188479
weight	5.10	0.196114
foreign	1.63	0.613765
Mean VIF	4.01	

Dakle, u našem modelu može postojati problem multikolinearnosti kod varijabli displacement i weight, s obzirom na to da su vrijednosti za VIF veće od 5. To ćemo detaljno ispitati u matrici korelacionih koeficijenata.

. regress mpg weight displacement foreign

. estat vce, correlation

Rezultat je sljedeći:

Correlation matrix of coefficients of **regress** model

e(V)	weight	displa~t	foreign	_cons
weight	1.0000			
displacement	-0.8352	1.0000		
foreign	0.1237	0.2316	1.0000	
_cons	-0.8099	0.3737	-0.5219	1.0000

Kako vidimo u matrici, visoke vrijednosti korelacionih koeficijenata su za varijable displacement i weight, te za koeficijent konstante i weight.

9.4 Komanda estat hettest

9.4.1 Opis

Komanda **estat hettest** koristi se za dobijanje vrijednosti Breusch-Pagan/ Cook-Weisberg testa za heteroskedastičnost regresionog modela. Ovim testovima ispitujemo pretpostavku postojanja konstantne varijanse greške. Homoskedastičnost znači da varijansa reziduala mora biti ista bez obzira na njihovu predviđenu vrijednost. Prisutnost heteroskedastičnosti (odnosno odsustvo homoskedastičnosti) kreira pristrasnost u predviđanju standardnih grešaka modela. Ova komanda izbacuje rezultat chi-kvadratnog testa nulte hipoteze koja pretpostavlja da postoji homoskedastičnost u modelu, pa prema tome ako je p-vrijednost manja od 0,05 to znači da u našem modelu imamo problem s heteroskedastičnosti i da predviđeni rezultati na osnovu našeg modela mogu biti pristrasni.

9.4.2 Sintaksa

Oblik sintakse jednak je nazivu komande, tj. estat hettest.

9.4.3 Primjer

Nastavljamo s istim modelom kao iz prethodnog primjera.

```
. regress mpg weight displacement foreign

. estat hettest

Breusch-Pagan / Cook-Weisberg test for heteroskedasticity
         Ho: Constant variance
         Variables: fitted values of mpg

         chi2(1)      =     8.01
         Prob > chi2  =   0.0046
```

Kao što vidimo iz prethodnog rezultata, naš model ima problema s heteroskedastičnošću jer je p-vrijednost niža od 0,05.

9.5 Komanda sktest

9.5.1 Opis

Komanda **sktest** koristi se za ispitivanje da li se vrijednosti koeficijenta simetričnosti i spljoštenosti značajno razlikuju u odnosu na one koji bi bili da je u pitanju normalna distribucija vrijednosti. Ovu komandu koristimo da ispitamo pretpostavku normalnog rasporeda grešaka. Postoji još načina za provjeru ove pretpostavke o čemu ćemo u primjeru.

9.5.2 Sintaksa

Osnovna sintaksa glasi: sktest varlist.

9.5.3 Primjer

Koristimo isti model kao u prethodnom primjeru i koristimo komandu predict kako bi dobili vrijednosti reziduala u varijabli čiji ćemo raspored vrijednosti ispitati.

```
. regress mpg weight displacement foreign

. predict res, residual

. sum res, detail
```

Residuals

	Percentiles	Smallest		
1%	-6.17923	-6.17923		
5%	-4.328157	-5.020157		
10%	-2.946146	-4.417861	Obs	74
25%	-2.073086	-4.328157	Sum of Wgt.	74
50%	-.4575869		Mean	-3.11e-09
		Largest	Std. Dev.	3.359183
75%	.7721555	7.936028		
90%	4.180274	8.273222	Variance	11.28411
95%	7.936028	8.453313	Skewness	1.72822
99%	14.39907	14.39907	Kurtosis	7.235684

Kako vidimo, koeficijent simetričnosti iznosi 1,72822 (kod normalne distribucije jednak je 0), a koeficijent spljoštenosti iznosi 7,235684 (kod normalne distribucije jednak je 3). I prema ovom ispisu vidimo da raspored vrijednosti u varijabli res ne prati oblik normalne distribucije, što možemo dodatno potvrditi i komadnom sktest.

```
. sktest res
```

Skewness/Kurtosis tests for Normality

Variable	Obs	Pr(Skewness)	Pr(Kurtosis)	joint adj chi2(2)	joint Prob>chi2
res	74	0.0000	0.0002	27.85	0.0000

I iz sktesta vidimo da se raspored vrijednosti reziduala značajno razlikuje od normalne distribucije, jer je test statistički značajan.

10. Metode uzorkovanja

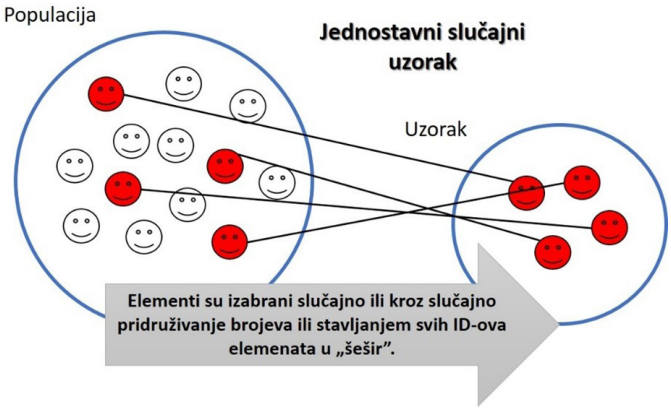
Za analizu masovnih pojava koje su predmet statističkih istraživanja najčešće se umjesto korištenja podataka iz cjelokupne populacije koriste podaci o manjoj skupini elemenata populacije koje nazivamo uzorkom. Razlog korištenja uzoraka jeste njihova efikasnost da uz manje podataka, a samim tim i manje troškove njihovog prikupljanja, donesemo statistički značajne zaključke o populaciji. Postoji više različitih metoda kojima se iz osnovnog skupa (populacije) biraju elementi koji će biti u uzorku.

Najveća podjela između metoda za izbor uzorka jeste na slučajno i namjerno odabrane uzorke. Osnovna razlika je u tome što kod slučajno odabranih uzoraka svaka jedinica osnovnog skupa ima jednaku mogućnost da bude izabrana za ispitivanje, odnosno unaprijed se može odrediti vjerovatnoća izbora svakog elementa u uzorak, dok je kod namjerno odabranih uzoraka izbor jedinica u uzorak subjektivan. U ovom Priručniku prezentujemo vam detaljnije 4 osnovna metoda za slučajni izbor elemenata u uzorak, uz prateće sintakse koje možete iskoristiti u programu Stata, kao i primjere istih.

10.1 Metode za slučajni izbor elemenata u uzorak

10.1.1 Jednostavni slučajni uzorak

Ovo je najčešći metod uzorkovanja kojeg karakteriše to da svaki element populacije ima jednaku vjerovatnoću da bude izabran u uzorak. Može se kreirati jednostavni slučajni uzorak s ili bez ponavljanja. Razlika je u tome što ako je uzorak izabran s ponavljanjem to znači da jedan element populacije može više puta biti izabran u uzorak. Češća je upotreba jednostavnog slučajnog uzorka bez ponavljanja, s obzirom na to da na taj način dobijamo pouzdanije rezultate. Kao primjer jednostavnog slučajnog uzorka možemo navesti izvlačenje loto brojeva ili izvlačenje imena iz šešira. Grafički to prikazujemo ovako:

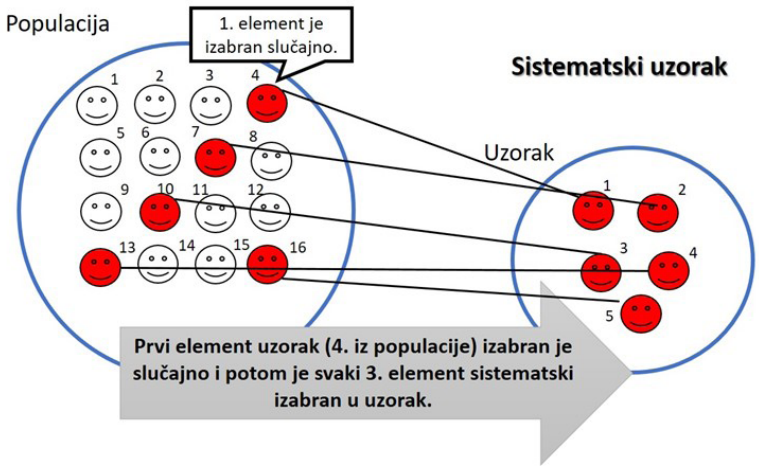


Slika 1: Jednostavni slučajni uzorak

Prednost korištenja jednostavnog slučajnog uzorka jeste njegova jednostavnost kod manjih populacija i to što daje visoko reprezentativan uzorak ciljne populacije. Nedostatak je korištenje kod velikih populacija, zbog toga što zahtijeva mnogo utrošenog vremena i novca, te se kao nedostatak može navesti i to da određeni manji segmenti u populaciji mogu biti izostavljeni upravo zbog osobine da je šansa izbora svakog elementa uzorka ista.

10.1.2 Sistematski uzorak

Ovaj metod uzorkovanja karakteriše to što se izbor elemenata vrši određenim sistematskim redom odabirajući slučajno početak. Jednostavnim sistematskim izborom se po redu broje elementi osnovnog skupa i za uzorak se odabere npr. svaki drugi, peti, k-ti element. Redni broj od kojeg počinje brojanje bira se slučajnim izborom. Treba napomenuti da je interval između dva elementa izabrana u uzorak isti za sve elemente. Dakle, bira se slučajno prvi element koji ulazi u uzorak a potom se određivanjem intervala definiše ostatak uzorka. Grafički sistematski uzorak prikazujemo ovako:

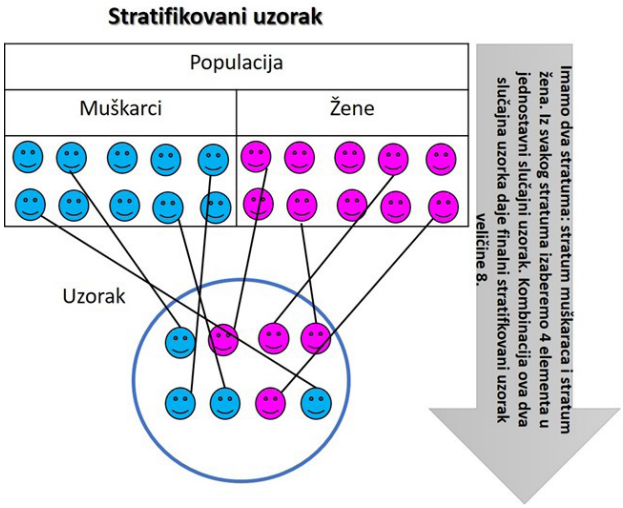


Slika 2: Sistematski uzorak

Osnovna prednost ove vrste uzoraka jeste što se uzorci formiraju prilično brzo i jednostavno, te ukoliko je osnovni skup dovoljno veliki, uzorak izabran na ovaj način je visoko reprezentativan predstavnik populacije. Najveći nedostatak ovog tipa uzorka je nereprezentativnost u slučajevima kada postoji određeni način prema kojem je populacija upisana u listu i ako se taj način poklapa s intervalom uzorka. Također može slično kao i jednostavni slučajni uzorak izostaviti određene manje segmente populacije koji mogu biti značajni.

10.1.3 Stratifikovani uzorak

Stratifikovani uzorak je specifična vrsta uzorka u kojem se isti kreira na način da se prvo cjelokupna populacija podjeli na manje podgrupe zvane stratumi. Jedinice koje čine jedan stratum moraju biti što homogenije, dok sami stratumi moraju biti što heterogeniji. Na ovaj način svaki stratum predstavlja jedan nezavisni osnovni skup. Potom u svakom stratumu možemo jednostavnim slučajnim uzorkovanjem odrediti elemente skupa koji ulaze u uzorak. Kombinovanjem više slučajnih uzoraka dobijenih iz startuma dobijamo finalni stratifikovani uzorak. Postoje dvije vrste stratifikovanog uzorka: proporcionalni i neproporcionalni. U proporcionalni uzorak biramo iz svakog stratuma broj elemenata jednak njegovom udjelu u populaciji, dok to nije slučaj s neproporcionalnim uzorkom. Na grafičkom prikazu stratifikovani uzorak izgleda ovako:

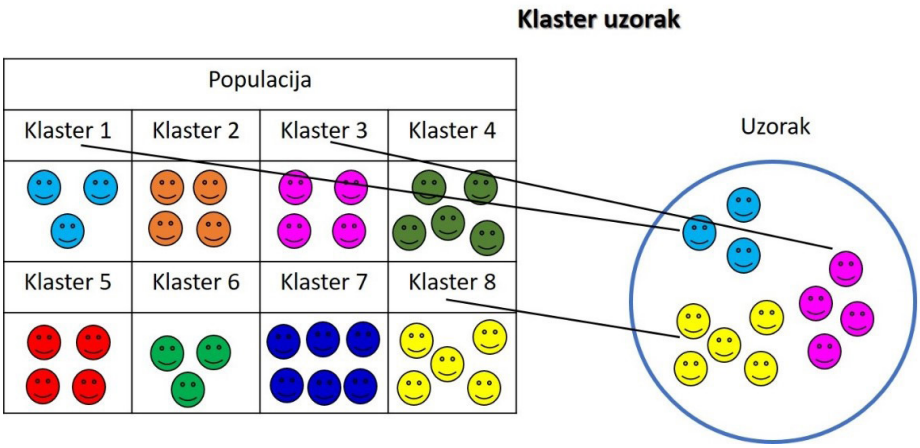


Slika 3: Stratifikovani uzorak

Korištenje stratifikovanog uzorka ima niz prednosti od kojih je najvažnija ta da se na ovaj način osigurava najveća preciznost u odnosu na druge metode uzorkovanja. Uzorak je reprezentativan s obzirom na to da omogućava uključivanje u uzorak određenih segmenata iz populacije koji nam mogu biti interesantni. Treba napomenuti da se u izboru elemenata u okviru jednog stratuma mogu koristiti i druge metode osim jednostavnog slučajnog uzorka. Nedostatak u korištenju stratifikovanog uzorka jeste njegova kompleksnost u fazi analize, kao i to što zahtijeva mnogo više napora i vremena u kreiranju uzorka i što postoji odvojen okvir za uzorkovanje svakog stratuma.

10.1.4 Klaster uzorak

U slučajevima kada je osnovni skup veliki i kada ne raspolažemo listom svih elemenata osnovnog skupa, moguće je koristiti klaster uzorkovanje. Osnovni skup se u ovom slučaju dijeli na prirodno određene klastere (skupine), te se potom slučajnim odabirom biraju klasteri koji će činiti uzorak. Svi elementi slučajno odabranog klastera ulaze u uzorak. Razlika između ovog tipa uzorka i stratifikovanog uzorka je činjenica da se u slučaju stratifikovanog uzorka izbor jedinica u uzorak vrši iz svakog stratuma. Grafički to izgleda ovako:



Slika 4: Klaster uzorak

Prednost korištenja klaster uzoraka je to što ne zahtijeva postojanje okvira uzorka i što je potrebno manje vremena i troškova prilikom uzorkovanja, te što ovi uzorci najčešće imaju veći broj elemenata. Kao nedostatak se može navesti da korištenje ove metode proizvodi više grešaka prilikom uzorkovanja (sampling error) i što je najmanje reprezentativan metod za prikaz populacije u odnosu na druge metode slučajnog uzorkovanja.

10.2 Komanda sample

10.2.1 Opis

sample povlači slučajne uzorke iz podataka u memoriji. Uzorkovanje je ovdje definisano kao povlačenje opservacija bez ponavljanja.

Veličina uzorka se može definisati kao procenat ili kao brojanje elemenata.

- sample bez opcije count povlači #% pseudoslučajnih uzoraka od podataka u memoriji uz odbacivanje (100-#)% opservacija

- sample s opcijom count povlači # opservacija pseudoslučajnih uzoraka od podataka u memoriji, pri tome odbacujući _N - #opservacija. # može biti i veći od _N, pri čemu su onda sve opservacije sačuvane u memoriji.

Ukoliko želite da se vaši rezultati mogu replicirati, morate prije korištenja komande sample koristiti komandu za uspostavljanje slučajnog početka brojanja **set seed**. Ova komanda će detaljnije biti objašnjenja u primjerima.

10.2.2 Sintaksa

Osnovna sintaksa ove komande glasi:

```
sample # [if][in][,count by(groupvars)]
```

Opcije koje su dopuštene su:

count određuje # u uzorku # i treba biti interpretirano kao brojanje opservacija prije nego određivanje procenata. Unosom sample 5 bez opcije count znači da će u uzorak biti povučeno 5% podataka iz memorije; unosom sample 5, count znači da će u uzorak biti povučeno 5 opservacija. Unos # većeg od broja opservacija u bazi podataka ne smatra se greškom.

by (groupvars) određuje koji #% uzorka će biti povučen iz svakog skupa grupne varijable, pri tome zadržavajući proporciju svake grupe.

count se može kombinovati s by(). Na primjer, unosom sample 50, count by(sex) kreirat će se uzorak od 50 muškaraca i 50 žena.

10.3 Primjeri prema metodama uzorkovanja

U nastavku ćemo prezentovati po jedan primjer u programu Stata za svaku gore navedenu metodu uzorkovanja. S obzirom na to da kreiranje uzoraka po određenim metodama podrazumijeva korištenje još nekoliko komandi pored komande sample, iste će biti objašnjenje uz prateće primjere.

10.3.1 Primjer – Jednostavni slučajni uzorak

Koristimo bazu auto.dta koja dolazi predinstalirana sa softverom Stata. Želimo kreirati uzorak jednostavnim slučajnim izborom elemenata veličine 10 jedinica. Koristimo sljedeće sintakse:

```
. gen id=_n
```

Ovu sintaksu upotrebljavamo kako bi svakoj opservaciji dodijeli redni broj. U ovoj bazi imamo ukupno 74 opservacije.

```
. keep make foreign id
```

Pošto baza ima više varijabli, na samom početku želimo zadržati samo one koje su nam neophodne, radi lakšeg prikaza elemenata uzorka. Koristimo komandu **keep** nakon koje definišemo nazive varijabli koje želimo zadržati. Mi smo odlučili zadržati varijable s nazivom automobila, porijeklom automobila i id brojem kojeg smo dodijelili svakom automobilu.

```
. set seed 10421
```

Kako je već ranije pomenuto, komanda set seed koristi se kako bi rezultati koje sintakse nakon ove komande proizvode bili isti kada vi replicirate sintakse. Inače, nakon navođenja komande upisujete bilo koji broj u kojem nema pravila u ponavljanju cifara.

```
. sample 10, count
```

S obzirom na to da nam treba 10 slučajno odabranih elemenata u uzorku, koristimo opciju count. Da nismo naveli opciju count dobili bismo uzorak veličine 7 (10% od 74 opservacija u bazi).

```
. list
```

Za prikaz elemenata uzorka koristimo komandu list. Dobijamo sljedeći ispis:

	make	foreign	id
1.	Plym. Arrow	Domestic	42
2.	Datsun 810	Foreign	59
3.	Olds 98	Domestic	35
4.	Cad. Deville	Domestic	11
5.	Dodge Colt	Domestic	20
6.	Olds Omega	Domestic	39
7.	AMC Spirit	Domestic	3
8.	Chev. Chevette	Domestic	14
9.	Mazda GLC	Foreign	63
10.	Olds Toronado	Domestic	41

10.3.2 Primjer –Sistematski uzorak

Ponovo učitamo bazu auto.dta. Želimo kreirati sistematski uzorak birajući svaki 4. automobil u bazi. To ćemo uraditi na sljedeći način (prve tri sintakse objašnjene su u prethodnom primjeru):

```
. gen id=_n

. keep make foreign id

. set seed 122

. di int(uniform()*4)+1

3
```

di je u programu Stata skraćenica za prikazivanje (display). Ovdje smo zapravo sračunali slučajni broj između 1 i 4 od kojeg će krenuti naš izbor elemenata. Dakle, dobili smo rezultat 3, što znači da će treći element osnovnog skupa biti prvi element našeg uzorka.

```
. drop if _n<3
```

Izbacujemo prve dvije opservacije (prva dva reda u programu Stata) korištenjem komande **drop**.

```
. gen newID=_n-1
```

Na ovaj način smo korištenjem komande **gen** kreirali novu varijablu koja će dodati redne brojeve preostalim elementima osnovnog skupa počevši od 0.

```
. gen y=mod(newID,4)
```

mod u programu Stata je skraćenica za **modulus**, odnosno označava početak nakon podjele. Ovdje smo kreirali novu varijablu koja daje vrijednost 0 za svaki 4. element osnovnog skupa.

```
. drop if y!=0
```

Potom smo izbacili sve opservacije čija vrijednost varijable y nije jednaka 0, tj. na taj način smo izbacili sve elemente skupa osim svakog 4. reda (elementa) s obzirom na to da smo s 0 označili svaki 4. element osnovnog skupa. Na ovaj način je kreiran uzorak biranjem svakog četvrtog automobila u bazi, a pri tome je početak od kojeg krećemo s brojanjem elemenata uzorka određen slučajnim izborom.

```
. list
```

	make	foreign	id	newID	y
1.	AMC Spirit	Domestic	3	0	0
2.	Buick Opel	Domestic	7	4	0
3.	Cad. Deville	Domestic	11	8	0
4.	Chev. Impala	Domestic	15	12	0
5.	Chev. Nova	Domestic	19	16	0
6.	Dodge St. Regis	Domestic	23	20	0
7.	Linc. Mark V	Domestic	27	24	0
8.	Merc. Marquis	Domestic	31	28	0
9.	Olds 98	Domestic	35	32	0
10.	Olds Omega	Domestic	39	36	0
11.	Plym. Champ	Domestic	43	40	0
12.	Pont. Catalina	Domestic	47	44	0
13.	Pont. Phoenix	Domestic	51	48	0
14.	BMW 320i	Foreign	55	52	0
15.	Datsun 810	Foreign	59	56	0
16.	Mazda GLC	Foreign	63	60	0
17.	Toyota Celica	Foreign	67	64	0
18.	VW Diesel	Foreign	71	68	0

10.3.3 Primjer – Stratifikovani uzorak

Učitavamo ponovo bazu auto.dta. Zadatak nam je da kreiramo stratifikovani uzorak od 8 automobila, po 4 domaća i strana automobila. Ponovit ćemo prve tri sintakse kao u prethodnim primjerima.

```
. gen id=_n

. keep make foreign id

. set seed 122

. sample 4, count by( foreign)
```

Korištenjem komande sample smo na ovaj način zadali programu Stata da po grupama varijable foreign slučajnim izborom uzme 4 elementa svake od grupa. U ovom slučaju su izabrana 4 domaća i 4 strana automobila u finalni uzorak, jer u varijabli foreign postoje samo 2 grupe.

```
. list
```

	make	foreign	id
1.	Linc. Continental	Domestic	26
2.	Ford Fiesta	Domestic	24
3.	AMC Pacer	Domestic	2
4.	Cad. Eldorado	Domestic	12
5.	Volvo 260	Foreign	74
6.	VW Diesel	Foreign	71
7.	VW Rabbit	Foreign	72
8.	Mazda GLC	Foreign	63

10.3.4 Primjer – Klaster uzorak

Zadatak nam je da kreiramo uzorak automobila od 2 slučajno izabrana klastera automobila. S obzirom na to da za klaster uzorak trebamo prirodno raspoređene elemente u klasterima, a radimo na bazi auto.dta, mi ćemo iskoristiti varijablu rep78 koja sadrži broj popravaka (repair record) automobila uz pretpostavku da ova podjela predstavlja prirodnu podjelu elemenata na klastere. Nakon učitavanja baze auto.dta, kreiranje klaster uzorka uradit ćemo na sljedeći način:

```
. contract rep78
```

Komanda **contract** u programu Stata kreira novi skup frekvencija određene varijable. Ovdje smo kreirali bazu podataka sa skupom frekvencija varijable rep78. Izbrojat ćemo frekvencije da prikažemo šta sada sadržava ova baza podataka.

```
. count

6
```

count prikazuje broj zapisa u trenutnoj memoriji. Kao što vidite, vidimo da varijabla rep78 ima 6 kategorija (klastera).

```
. set seed 10236

. sample 2, count
```

Na ovaj način dajemo programu Stata instrukciju da od 6 ponuđenih klastera slučajnim izborom izabere 2 klastera.

```
. sort rep78

. keep rep78
```

Korištenjem ove dvije sintakse sortirali smo vrijednosti varijable rep78 i u bazi zadržali samo varijablu rep78.

Potom spasimo ovu bazu podataka pod nazivom rep78. Zatvorimo program Stata ili očistimo memoriju korištenjem komande **clear all**. Učitamo ponovo bazu auto.dta. Ponovimo skup koraka kao u prethodnim primjerima:

```
. gen id=_n

. keep make rep78 foreign id
```

Razlika u odnosu na prethodne primjere je to što smo zadržali i varijablu rep78.

```
. sort rep78
```

Sortiramo sve opservacije prema elementima varijable rep78.

```
. merge m:m rep78 using “C: \.....\rep78.dta”
```

Spajamo dvije baze podataka korištenjem opcije **merge**. S obzirom na to da imamo nejednak broj opservacija u dvije baze, koristimo opciju m:m. Rezultat koji Stata ispisuje je sljedeći:

Result	# of obs.	
not matched	42	
from master	42	(_merge==1)
from using	0	(_merge==2)
matched	32	(_merge==3)

. drop if _merge!=3

Izbacujemo iz baze sve elemente čija oznaka nije jednaka 3. Kako vidimo iz gornjeg ispisa, imamo 32 elementa čija oznaka iz spajanja je jednaka 3. Dakle, brisanjem izbacujemo 42 opservacije.

. list

	make	rep78	foreign	id	_merge
1.	Pont. Firebird	1	Domestic	48	matched (3)
2.	Olds Starfire	1	Domestic	40	matched (3)
3.	Chev. Chevette	3	Domestic	14	matched (3)
4.	Olds Cutlass	3	Domestic	37	matched (3)
5.	AMC Pacer	3	Domestic	2	matched (3)
6.	Olds Cutl Supr	3	Domestic	36	matched (3)
7.	Merc. Zephyr	3	Domestic	34	matched (3)
8.	Merc. Marquis	3	Domestic	31	matched (3)
9.	Buick Riviera	3	Domestic	9	matched (3)
10.	Pont. Grand Prix	3	Domestic	49	matched (3)
11.	Linc. Versailles	3	Domestic	28	matched (3)
12.	AMC Concord	3	Domestic	1	matched (3)
13.	Linc. Mark V	3	Domestic	27	matched (3)
14.	Pont. Le Mans	3	Domestic	50	matched (3)
15.	Buick Skylark	3	Domestic	10	matched (3)
16.	Chev. Nova	3	Domestic	19	matched (3)
17.	Merc. Monarch	3	Domestic	32	matched (3)
18.	Buick LeSabre	3	Domestic	6	matched (3)
19.	Plym. Arrow	3	Domestic	42	matched (3)
20.	Plym. Horizon	3	Domestic	44	matched (3)

21.	Olds Omega	3	Domestic	39	matched (3)
22.	Chev. Malibu	3	Domestic	16	matched (3)
23.	Cad. Deville	3	Domestic	11	matched (3)
24.	Cad. Seville	3	Domestic	13	matched (3)
25.	Audi Fox	3	Foreign	54	matched (3)
26.	Linc. Continental	3	Domestic	26	matched (3)
27.	Renault Le Car	3	Foreign	65	matched (3)
28.	Olds Toronado	3	Domestic	41	matched (3)
29.	Buick Century	3	Domestic	4	matched (3)
30.	Ford Mustang	3	Domestic	25	matched (3)
31.	Buick Regal	3	Domestic	8	matched (3)
32.	Fiat Strada	3	Foreign	60	matched (3)

AKADEMIJA

ZA
UNAPREĐENJE ISTRAŽIVANJA TRŽIŠTA RADA

Projekat implementira:

NIRAS GOPA WORLDWIDE CONSULTANTS pôle emploi