DEO I

Istraživanje

Cilj prvog dela ove knjige je da što pre počnete da radite sa osnovnim alatkama za *istraživanje podataka* (engl. *data exploration*). Istraživanje podataka je umeće sagledavanja podataka, brzog postavljanja hipoteza, njihovog brzog proveravanja, a zatim višekratnog ponavljanja celog postupka. Cilj istraživanja podataka je generisanje mnogih obećavajućih smernica koje kasnije možete dublje proučavati.

U ovom delu knjige upoznaćete neke alatke i tehnike od kojih ćete odmah imati koristi:

- Vizuelizacija je odlična za započinjanje programiranja u R-u zato što je korist od nje sasvim očigledna: pravite elegantne i informativne dijagrame koji vam pomažu da shvatite podatke. U poglavlju 1 uranjate u vizuelno predstavljanje podataka, upoznajete osnovnu strukturu dijagrama koji se prave pomoću paketa **ggplot2** i učite moćne tehnike za pretvaranje podataka u grafičke prikaze (dijagrame).
- Sama vizuelizacija obično nije dovoljna, pa ćete u poglavlju 3 naučiti ključne glagole koji vam omogućavaju da izaberete važne promenljive, izdvojite najbitnije opservacije, definišete nove promenljive i izračunate sažetke podataka (engl. *summaries*).
- Konačno, u poglavlju 5, kombinovaćete vizuelizaciju i transformisanje sa svojom radoznalošću i skepticizmom da biste postavljali zanimljiva pitanja o podacima i odgovarali na njih.

Modelovanje je važan deo procesa istraživanja, ali još ne vladate veštinama potrebnim da biste ga efikasno savladali i primenili. Vratićemo mu se u delu IV, kada naučite više o alatkama za obradu podataka i programiranje.

Među pomenuta tri poglavlja o istraživanju podataka, smeštena su i tri poglavlja posvećena radnom toku (engl. *workflow*) u programskom okruženju R. U poglavljima 2, 4 i 6 upoznaćete dobre praktične tehnike za pisanje i organizovanje R koda. To će vas osposobiti za uspešan rad na duže staze, jer ćete umeti da primenite alatke koje obezbeđuju organizovanost u radu na stvarnim projektima.

Vizuelizacija podataka pomoću paketa ggplot2

Uvod

> „Jednostavan dijagram prenosi analitičaru podataka više informacija od bilo kog drugog sredstva."
>
> – John Tukey

U ovom poglavlju naučićete da grafički predstavite svoje podatke koristeći paket **ggplot2**. R ima nekoliko sistema za izradu dijagrama, ali je **ggplot2** jedan od najelegantnijih i najsvestranijih. **ggplot2** implementira *gramatiku dijagrama* (engl. *grammar of graphics*) – koherentan sistem za opisivanje i izradu grafikona. Uz **ggplot2** možete brže obaviti više posla, tako što ćete naučiti jedan sistem i primenjivati ga na mnogo mesta.

Ako želite da saznate više o teoriji na kojoj se zasniva **ggplot2** pre nego što počnete da ga koristite, preporučujemo da pročitate članak „A Layered Grammar of Graphics" (*http://vita.had.co.nz/papers/layered-grammar.pdf*).

Preduslovi

Fokus ovog poglavlja je **ggplot2** – jedan od osnovnih elemenata jezgra paketa tidyverse. Da biste pristupili skupovima podataka, stranicama s pomoćnim informacijama i funkcijama koje ćemo koristiti u ovom poglavlju, učitajte paket tidyverse tako što ćete izvršiti sledeći kôd:

```
library(tidyverse)
#> Loading tidyverse: ggplot2
#> Loading tidyverse: tibble
#> Loading tidyverse: tidyr
#> Loading tidyverse: readr
#> Loading tidyverse: purrr
#> Loading tidyverse: dplyr
#> Conflicts with tidy packages ----------------------------------------------
#> filter(): dplyr, stats
#> lag():    dplyr, stats
```

Ovaj jedan red koda učitava jezgro paketa tidyverse – pakete koje ćete koristiti u gotovo svakoj analizi podataka. On vam, takođe, govori koje su funkcije iz paketa tidyverse u sukobu s funkcijama iz osnovnog R-a (ili iz drugih paketa koje ste možda učitali).

Ako izvršite navedeni kôd i dobijete poruku o grešci „there is no package called 'tidyverse'", morate prvo instalirati paket tidyverse, a zatim ponovo izvršiti komandu `library()`.

```
install.packages("tidyverse")
library(tidyverse)
```

Paket instalirate samo jednom, ali ga morate ponovo učitati kad god započinjete novu sesiju.

Ukoliko treba eksplicitno da navedemo odakle potiče neka funkcija (ili skup podataka), koristimo poseban oblik komande: `paket::funkcija()`. Na primer, `ggplot2::ggplot()` vam eksplicitno govori da koristimo funkciju `ggplot()` iz paketa **ggplot2**.

Prvi koraci

Iskoristimo prvi dijagram da bismo odgovorili na pitanje: Da li automobili s motorima velikih zapremina troše više goriva od automobila s motorima malih zapremina? Verovatno već imate odgovor, ali pokušajte da on bude precizan. Kako izgleda odnos između veličine motora i potrošnje goriva? Da li je on pozitivan? Negativan? Linearan? Nelinearan?

Okvir s podacima mpg

Odgovor možete da testirate pomoću *okvira s podacima* (engl. *data frame*) `mpg` iz paketa **ggplot2** (tj. `ggplot2::mpg`). Okvir s podacima je pravougaoni skup promenljivih (u kolonama) i opservacija (u redovima). `mpg` sadrži opservacije koje je prikupila američka organizacija za zaštitu životne sredine (Environment Protection Agency) za 38 modela automobila.

```
mpg
#> # Tibl: 234 × 11
#>   manufacturer model displ  year   cyl      trans   drv   cty   hwy    fl
#>          <chr> <chr> <dbl> <int> <int>      <chr> <chr> <int> <int> <chr>
#> 1         audi    a4   1.8  1999     4   auto(l5)     f    18    29     p
#> 2         audi    a4   1.8  1999     4 manual(m5)     f    21    29     p
#> 3         audi    a4   2.0  2008     4 manual(m6)     f    20    31     p
#> 4         audi    a4   2.0  2008     4   auto(av)     f    21    30     p
#> 5         audi    a4   2.8  1999     6   auto(l5)     f    16    26     p
#> 6         audi    a4   2.8  1999     6 manual(m5)     f    18    26     p
#> # ... i još 228 redova i jedna dodatna promenljiva: class <chr>
```

Među promenljivama u okviru `mpg` nalaze se i sledeće:

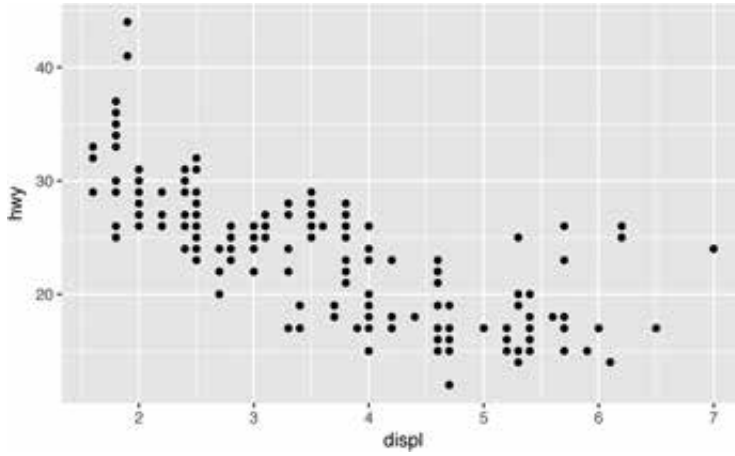
- `displ`, zapremina motora automobila, u litrima.
- `hwy`, efikasnost potrošnje goriva na autoputu, u miljama po galonu (mpg). Automobil s niskom efikasnošću potrošnje goriva troši više goriva od automobila s visokom efikasnošću potrošnje kada prelaze isto rastojanje.

Da biste saznali više o mpg podacima, otvorite pripadajuću stranu s pomoćnim informacijama (engl. *help page*) tako što ćete zadati komandu ?mpg.

Izrada ggplot dijagrama

Da biste grafički prikazali mpg podatke, izvršite sledeći kôd kako biste postavili promenljivu displ na x-osu a hwy na y-osu:

```
ggplot(data = mpg) +
  geom_point(mapping = aes(x = displ, y = hwy))
```



Dijagram prikazuje negativnu vezu između zapremine motora (displ) i efikasnosti potrošnje goriva (hwy). Drugim rečima, automobili s većim motorima troše više goriva. Da li to potvrđuje ili negira vašu hipotezu o efikasnosti potrošnje goriva i zapremini motora?

Kada koristite **ggplot2**, crtanje dijagrama započinjete funkcijom ggplot(). ggplot() crta koordinatni sistem kome možete dodavati slojeve. Prvi argument funkcije ggplot() jeste skup podataka koji će se koristiti na dijagramu. Znači, ggplot(data = mpg) crta prazan dijagram, ali on nije baš zanimljiv, pa ga ovde nećemo prikazati.

Dijagram ćete dovršiti tako što ćete dodati jedan ili više slojeva (engl. *layers*) funkciji ggplot(). Funkcija geom_point() dodaje dijagramu sloj tačaka, pa dobijate dijagram rasturanja (engl. *scatterplot*). Paket **ggplot2** sadrži mnoge geom funkcije, od kojih svaka dijagramu dodaje drugačiju vrstu sloja. O mnogima od njih učićete u ovom poglavlju.

Svaka geom funkcija iz paketa **ggplot2** prihvata argument mapping. On definiše kako se promenljivama iz vašeg skupa podataka pridružuju (engl. *map*) vizuelna svojstva. Argument mapping je uvek uparen s funkcijom aes(), a argumenti x i y funkcije aes() određuju koje promenljive treba pridružiti x i y osama. **ggplot2** traži pridruženu promenljivu u argumentu data – u ovom sučaju, to je mpg.

Šablon za izradu dijagrama

Pretvorimo ovaj kôd u višekratno upotrebljiv šablon (engl. *reusable template*) za izradu dijagrama pomoću paketa **ggplot2**. Da biste nacrtali dijagram, zamenite delove koda u uganim zagradama (< i >) skupom podataka, geom funkcijom ili skupom pridruženih parova.

```
ggplot(data = <DATA>) +
  <GEOM_FUNCTION>(mapping = aes(<MAPPINGS>))
```

U ostatku poglavlja saznaćete kako da dovršite i proširite ovaj šablon da biste crtali različite vrste dijagrama. Počećemo od komponente `<MAPPINGS>`.

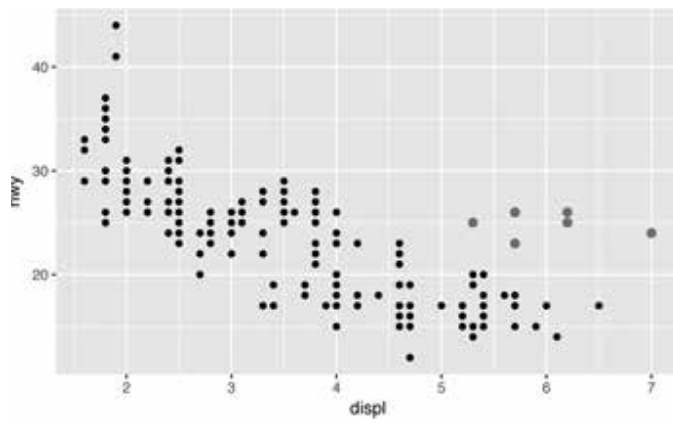
Vežbe

1. Kada izvršite komandu `ggplot(data = mpg)`, šta vidite?
2. Koliko ima redova u `mpg`? Koliko kolona?
3. Šta opisuje promenljiva `drv`? Da biste to saznali, pročitajte pomoćne informacije za `?mpg`.
4. Napravite dijagram rasturanja `hwy` u zavisnosti od `cyl`.
5. Šta se dešava ako nacrtate dijagram rasturanja `class` u zavisnosti od `drv`. Zašto taj dijagram nije koristan?

Pridruživanje estetskih svojstava

> „Najveća vrednost slike je kada nas primora da zapazimo nešto što nikada nismo očekivali da vidimo."
>
> – John Tukey

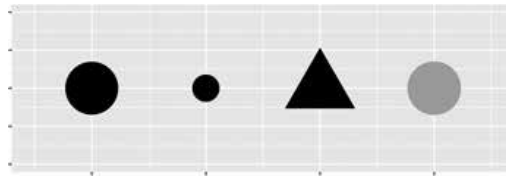
Na narednom dijagramu izgleda da se jedna grupa tačaka (istaknutih crvenom bojom) nalazi van linearnog trenda. Ti automobili prelaze duži put nego što očekujete uz datu potrošnju goriva. Kako možete da objasnite te slučajeve?



U BOJI NA KRAJU KNJIGE

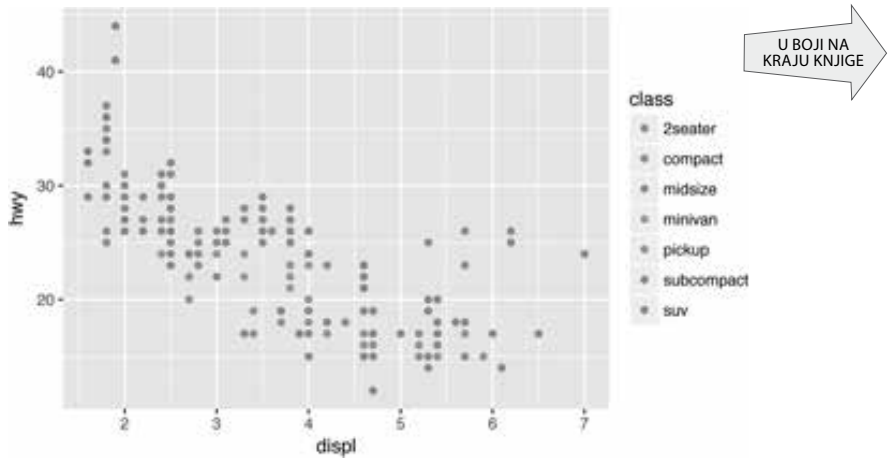
Pretpostavimo da su ti automobili hibridi. Jedan od načina da proverimo tu hipotezu jeste da pogledamo vrednost promenljive `class` za svaki automobil. Promenljiva `class` iz skupa podataka `mpg` svrstava automobile u grupe (klase) kao što su kompaktni, srednje klase i SUV. Ako prikazane netipične tačke (engl. *outlying points*) predstavljaju hibride, oni bi trebalo da su klasifikovani kao kompaktni automobili ili, možda, niža klasa, engl. *subcompact cars* (imajte na umu da su ovi podaci sakupljeni pre nego što su hibridni kamioneti i SUV vozila postali popularni).

Dvodimenzionalnom dijagramu rasturanja možete dodati treću promenljivu – recimo, `class` – tako što ćete je pridružiti promenljivoj (mapirati) kao *estetsko svojstvo* (engl. *aesthetic*). Estetsko svojstvo je vizuelna karakteristika objekata na dijagramu. U estetska svojstva spadaju – na primer – veličina, oblik ili boja tačaka. Tačku (kao što je ona na sledećoj slici) možete prikazati na različite načine tako što ćete promeniti vrednosti njenih estetskih svojstava. Pošto već koristimo reč „vrednost“ (engl. *value*) da opišemo podatke, koristićemo reč „nivo“ (engl. *level*) za opisivanje estetskih svojstava. Ovde menjamo nivoe veličine, oblika i boje tačke da bismo dobili malu, trouglastu ili sivu tačku:



Informacije koje se odnose na podatke možete preneti posmatračima tako što ćete estetska svojstva na svom dijagramu pridružiti promenljivama u korišćenom skupu podataka. Na primer, boje tačaka možete pridružiti promenljivoj `class` da biste ukazali na klasu svakog automobila.

```
ggplot(data = mpg) +
  geom_point(mapping = aes(x = displ, y = hwy, color = class))
```



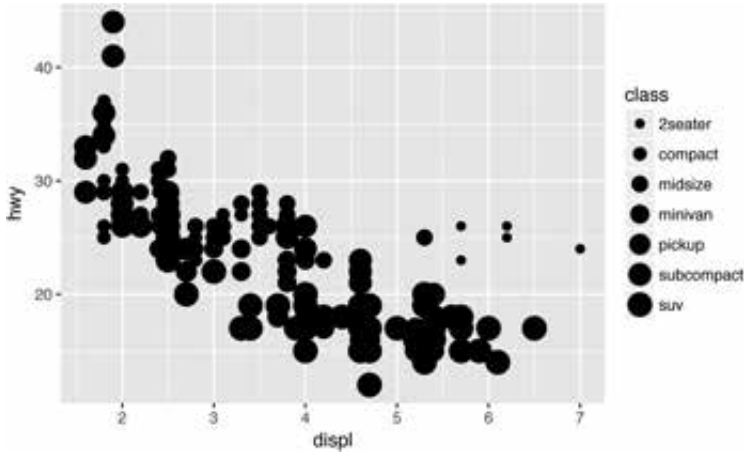
(Ako više volite britansku varijantu engleskog jezika – kao Hadley – možete pisati `colour` umesto `color`.)

Da biste promenljivoj pridružili estetsko svojstvo, vežite ime svojstva za ime promenljive unutar funkcije `aes()`. **ggplot2** će automatski dodeliti jedinstven nivo tog estetskog svojstva (u ovom slučaju, jedinstvenu boju) svakoj jedinstvenoj vrednosti te promenljive – što je proces poznat kao *skaliranje* (engl. *scaling*). Osim toga, **ggplot2** će dodati i legendu koja objašnjava koji nivoi odgovaraju kojim vrednostima.

Boje otkrivaju da se mnoge od neobičnih tačaka odnose na automobile dvosede. Oni ne spadaju u hibride već su, u stvari, sportski automobili! Sportski automobili imaju motore velike zapremine – kao SUV vozila i kamioneti – ali su malih gabarita, poput vozila srednje klase ili kompaktnih automobila, pa efikasnije troše gorivo. Iz ove perspektive, bilo je i nelogično da ti automobili budu hibridi pošto imaju motore velike zapremine.

U navedenom primeru, promenljivoj `class` pridružili smo boju, ali smo na isti način mogli da joj pridružimo i neko drugo estetsko svojstvo – recimo, veličinu. U tom slučaju, tačna veličina svake tačke otkrila bi kojoj je klasi data tačka dodeljena. Međutim, dobićemo *upozorenje* (engl. *warning*), pošto nije dobro neuređenu (engl. *unordered*) promenljivu (`class`) pridružiti uređenom (engl. *ordered*) estetskom svojstvu (`size`).

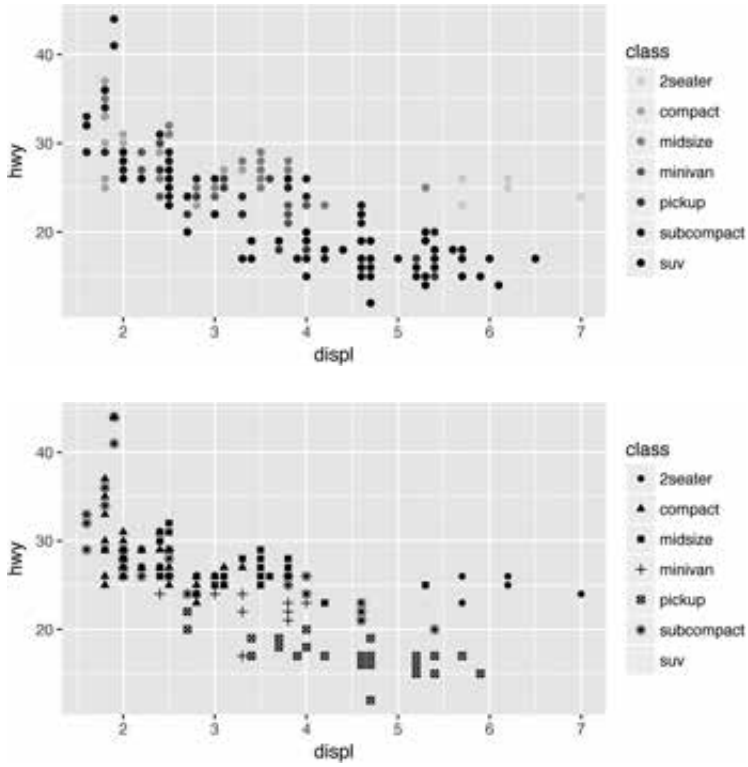
```
ggplot(data = mpg) +
  geom_point(mapping = aes(x = displ, y = hwy, size = class))
#> Warning: Using size for a discrete variable is not advised.
```



Promenljivoj `class` mogli smo pridružiti i estetsko svojstvo *alpha*, koje upravlja providnošću tačaka, ili pak oblik tačaka.

```
# Gore
ggplot(data = mpg) +
  geom_point(mapping = aes(x = displ, y = hwy, alpha = class))
```

```
# Dole
ggplot(data = mpg) +
  geom_point(mapping = aes(x = displ, y = hwy, shape = class))
```



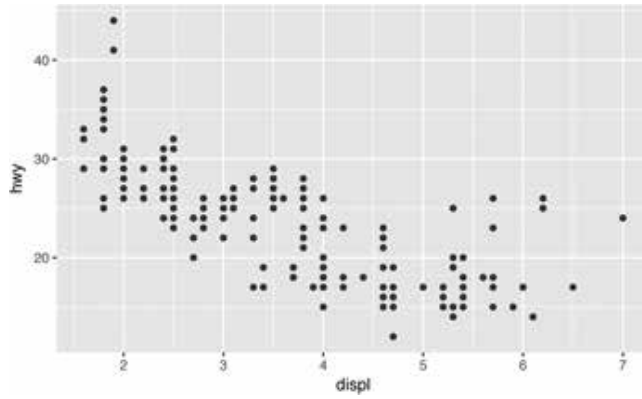
Šta se dogodilo sa SUV vozilima? **ggplot2** koristi samo šest oblika istovremeno. Kada kao estetsko svojstvo koristite oblik, dodatne grupe se podrazumevano neće prikazati na dijagramu.

Za svako estetsko svojstvo koristite funkcije `aes()` da biste ime datog svojstva vezali za promenljivu koju prikazujete na dijagramu. Funkcija `aes()` prikuplja sva pridruživanja estetskih svojstava koja se koriste na određenom sloju i prosleđuje ih argumentu sloja koji se odnosi na pridruživanje. Navedena sintaksa sadrži i korisne informacije o `x` i `y`: x i y položaji tačke sami su po sebi estetska svojstva, tj. vizuelna svojstva koja možete pridružiti promenljivama da biste prikazali informacije o datim podacima.

Nakon što ste promenljivoj pridružili estetsko svojstvo, **ggplot2** će se pobrinuti za sve ostalo. On bira skalu koja odgovara datom estetskom svojstvu i pravi legendu koja objašnjava vezu između nivoa i vrednosti. Za estetska svojstva x i y, **ggplot2** ne pravi legendu, ali crta liniju ose, s podeocima (engl. *tick marks*) i natpisom (engl. *label*). Ta linija služi kao legenda – objašnjava vezu između položaja i vrednosti.

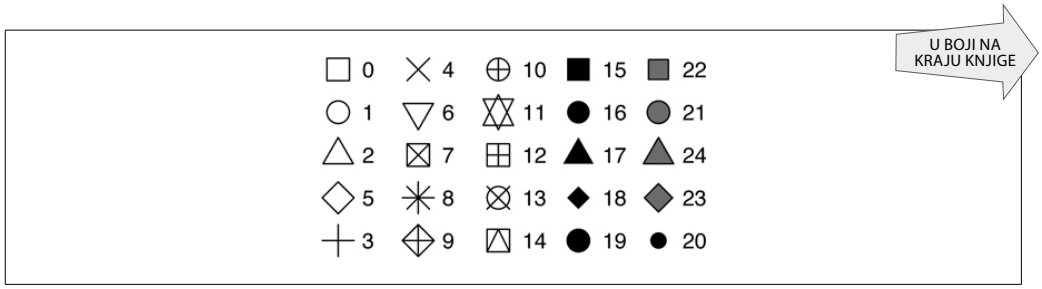
Estetska svojstva geom funkcije možete *zadati* i ručno. Na primer, možemo zadati da sve tačke na dijagramu budu plave:

```
ggplot(data = mpg) +
  geom_point(mapping = aes(x = displ, y = hwy), color = "blue")
```



U ovom slučaju, boja ne prenosi informaciju o promenljivoj već samo menja izgled dijagrama. Da biste ručno zadali estetsko svojstvo, navedite njegovo ime kao argument odgovarajuće geom funkcije; tj. ono treba da bude *izvan* funkcije `aes()`. Morate izabrati vrednost koja ima smisla za to estetsko svojstvo:

- Ime boje kao znakovni niz (engl. *character string*).
- Veličinu tačke u mm.
- Oblik tačke kao broj, prema slici 1.1. Neki oblici deluju kao da su duplikati: na primer, 0, 15 i 22 – svi su kvadrati. Razlika je u interakciji estetskih svojstava `colour` i `fill`. Šuplji oblici (0–14) imaju ivicu određenu svojstvom `colour`; puni oblici (15–18) popunjeni su bojom `colour`; popunjeni oblici (21–24) imaju ivicu u boji `colour` i popunu `fill`.

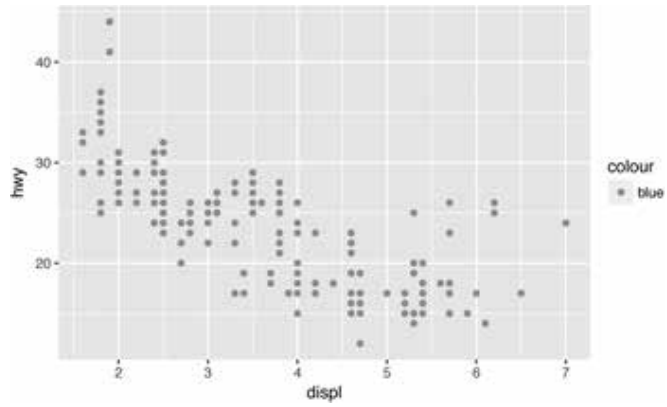


Slika 1-1. R ima 25 ugrađenih oblika označenih brojevima.

Vežbe

1. Šta nije u redu sa ovim kodom? Zašto tačke nisu plave?

```
ggplot(data = mpg) +
  geom_point(
    mapping = aes(x = displ, y = hwy, color = "blue")
  )
```



U BOJI NA KRAJU KNJIGE

2. Koje su promenljive u skupu podataka `mpg` kategorijske? Koje promenljive su neprekidne (kontinualne)? (Pomoć: otkucajte `?mpg` da biste pročitali dokumentaciju o tom skupu podataka). Kako možete videti te informacije kada izvršite `mpg`?
3. Neprekidnoj promenljivoj pridružite svojstva `color`, `size` i `shape`. Po čemu se razlikuje ponašanje ovih estetskih svojstava za kategorijske i neprekidne promenljive?
4. Šta se dešava ako istoj promenljivoj pridružite više estetskih svojstava?
5. Kako deluje estetsko svojstvo `stroke`? S kojim oblicima je ono primenjivo? (Pomoć: upotrebite `?geom_point`.)
6. Šta se događa ako neko estetsko svojstvo pridružite nečem drugom a ne imenu promenljive – na primer, `aes(colour = displ < 5)`?

Uobičajeni problemi

Kad počnete da izvršavate R kôd, verovatno ćete naići na probleme. Ne brinite – to se dešava svima. Pišem R kôd već godinama, a ipak svakog dana napišem i kôd koji ne radi!

Počnite tako što ćete pažljivo uporediti kôd koji izvršavate s kodom iz ove knjige. R je ekstremno izbirljiv, pa i znak na pogrešnom mestu može napraviti problem. Uverite se da za svaku levu zagradu postoji i desna, a i da su navodnici upareni. Ponekad ćete pokrenuti kôd i neće se desiti ništa. Pogledajte desnu stranu konzole: ako vidite znak +, R smatra da niste uneli ceo izraz i čeka da ga dovršite. U takvom slučaju, obično je lakše da počnete od početka tako što ćete pritisnuti Esc da biste obustavili obradu tekuće komande.