R algajatele,
eriti neile kes pole varem programmeerinud

Ott Toomet

Tartu, 9. mai 2006

Kava

1. Taust
2. Üldist
3. Muutujad
4. Andmed
5. Indekseerimine
6. Arvutamine
7. Funktsioon
8. Statistika
9. Graafika

Taust

Mis on R?

- S keele variant
- Üldotstarbeline kõrgtaseme programmeerimiskeel
- Statistikale orienteeritud vahendid
  - vektorid
  - maatriksid
  - statistilised meetodid
- vabavara (GPL)
- Väga paindlik
  - Uued statistilised meetodid
  - Andmetöötlus
  - Kompileeritud kood (C, fortran)

Taust

Mis on R?

- S keele variant
- Üldotstarbeline kõrgtaseme programmeerimiskeel
- Statistikale orienteeritud vahendid
  - vektorid
  - maatriksid
  - statistilised meetodid
- vabavara (GPL)
- Väga paindlik
  - Uued statistilised meetodid
  - Andmetöötlus
  - Kompileeritud kood (C, fortran)

Taust

Mis on R?

- S keele variant
- Üldotstarbeline kõrgtaseme programmeerimiskeel
- Statistikale orienteeritud vahendid
  - vektorid
  - maatriksid
  - statistilised meetodid
- vabavara (GPL)
- Väga paindlik
  - Uued statistilised meetodid
  - Andmetöötlus
  - Kompileeritud kood (C, fortran)

Paigaldamine

- Valmis *binary*-paketid
- Lähtekood
- Paigalda ka R-i toetusega tekstiredaktor!

Kasutuskeskkond

- Käsurida, interaktiivne
  - Nõuab asjast aru saamist
- *batch*-programm süsteemi käsurealt, automaatne
- ESS
- GUI-d
- Sisseehitatud help (html ja käsurealt)
- R-list

Kasutuskeskkond

- Käsurida, interaktiivne
  - Nõuab asjast aru saamist
- *batch*-programm süsteemi käsurealt, automaatne
- ESS
- GUI-d
- Sisseehitatud help (html ja käsurealt)
- R-list

Kasutuskeskkond

- Käsurida, interaktiivne
  - Nõuab asjast aru saamist
- *batch*-programm süsteemi käsurealt, automaatne
- ESS
- GUI-d
- Sisseehitatud help (html ja käsurealt)
- R-list

Paketid

- Modulaarne struktuur
- Lisapaketid igasugu lisafunktsioonide tarvis
- CRAN
- `library()`
- `install.packages()`

Käsurida ja programm

R-i juhitakse tavaliselt käskuda abil

- väga võimalusterohke
  10 sümboliline käsk (70 eri sümbolit) $\sim 10^{18}$ eri kombinatsiooni
- Lihtne eelnevaid käske korrata ja modifitseerida
- Raskem meelde jätta

Salvestatud käskude jada on programm

- Kasulik kirjutada programm kui ≥ 3 käsurida
- Rutiinsed andmetöötluse operatsioonid
- Kõik mida vaja uuesti kasutada ja modifitseerida

Programmi saab käivitada *batch*-programmina

Käsurida ja programm

R-i juhitakse tavaliselt käskuda abil

- väga võimalusterohke
  10 sümboliline käsk (70 eri sümbolit) $\sim 10^{18}$ eri kombinatsiooni
- Lihtne eelnevaid käske korrata ja modifitseerida
- Raskem meelde jätta

Salvestatud käskude jada on programm

- Kasulik kirjutada programm kui ≥ 3 käsurida
- Rutiinsed andmetöötluse operatsioonid
- Kõik mida vaja uuesti kasutada ja modifitseerida

Programmi saab käivitada *batch*-programmina

Muutujad 1

- Muutuja on nimi mälus olevale infole.
- Muutuja võib sisaldada:
  - Arve (*numeric*: *integer* ning *double*)
  - Teksti (*character*)
  - Loogilisi väärtusi *TRUE* ja *FALSE* (*logical*)
  - Muud, s.h. funktsioone, avaldisi, internetiühendusi...
- Kategooriline muutuja: *factor*
- Puuduv väärtus: *NA* ja *NaN*
- Attribuudid, nimed
- Klassid

Muutujad 1

- Muutuja on nimi mälus olevale infole.
- Muutuja võib sisaldada:
  - Arve (*numeric*: *integer* ning *double*)
  - Teksti (*character*)
  - Loogilisi väärtusi *TRUE* ja *FALSE* (*logical*)
  - Muud, s.h. funktsioone, avaldisi, internetiühendusi...
- Kategooriline muutuja: *factor*
- Puuduv väärtus: *NA* ja *NaN*
- Attribuudid, nimed
- Klassid

Muutujad 1

- Muutuja on nimi mälus olevale infole.
- Muutuja võib sisaldada:
  - Arve (*numeric*: *integer* ning *double*)
  - Teksti (*character*)
  - Loogilisi väärtusi *TRUE* ja *FALSE* (*logical*)
  - Muud, s.h. funktsioone, avaldisi, internetiühendusi...
- Kategooriline muutuja: *factor*
- Puuduv väärtus: *NA* ja *NaN*
- Attribuudid, nimed
- Klassid

Muutujad 2

- Kõik muutujad on vektorid
- Maatriksid
- Eri tüüpi muutujad ühes tükis: *list*
  - Listi komponentidel (muutujatel) nimed
- Listi erijuht: andmebaas (*data frame*):
  - Kõigil muutujatel ühepalju vaatlusi
  - Kõigil muutujatel nimed
- Konverteerimine: as.*tüüp*

Muutujad 2

- Kõik muutujad on vektorid
- Maatriksid
- Eri tüüpi muutujad ühes tükis: *list*
  - Listi komponentidel (muutujatel) nimed
- Listi erijuht: andmebaas (*data frame*):
  - Kõigil muutujatel ühepalju vaatlusi
  - Kõigil muutujatel nimed
- Konverteerimine: `as.`*tüüp*

Muutujad 3

Defineeritud muutujad:

- `pi` = 3,14
- `T` = `TRUE`
- `F` = `FALSE`

Teeme ise andmed

- `c()`–funktsioon
- Jadad, `seq()`
- `rep()`–funktsioon

Failiformaadid

- R andmefailid (.Rdat)
  - `load()`, `save()`
- ASCII failid (.csv jne)
- *foreign*-library toetab
  - SPSS (.sav)
  - STATA (.dta)
- Kõige universaalsem on ASCII esitus
- Kõige kiirem on R enda failid

Failiformaadid

- R andmefailid (.Rdat)
  - `load()`, `save()`
- ASCII failid (.csv jne)
- *foreign*-library toetab
  - SPSS (.sav)
  - STATA (.dta)
- Kõige universaalsem on ASCII esitus
- Kõige kiirem on R enda failid

Failiformaadid

- R andmefailid (.Rdat)
  - `load()`, `save()`
- ASCII failid (.csv jne)
- *foreign*-library toetab
  - SPSS (.sav)
  - STATA (.dta)
- Kõige universaalsem on ASCII esitus
- Kõige kiirem on R enda failid

Indekseerimine

Kuidas vektori/maatriksi vajalikku elementi näperdada

- `length()` – vektori pikkus
- `dim()` – maatriksi mõõtmed
- Erinevad indeksi tüübid:
  - Täisarvud: `v[c(1,2,5)]`
  - Negatiivsed täisarvud: `v[-1]`
  - Loogiline indeks: `v[c(T, F, T)]`
  - Komponentide nimed: `v[c("beta", "gamma")]`
  - Kõik komponendid: `v[]`
- Vajalikele elementidele omistamine:
  `v[v < 0] <- 0`
- Andmebaasist selekteerimine:
  `data[data$income > 0,]`

Indekseerimine

Kuidas vektori/maatriksi vajalikku elementi näperdada

- `length()` – vektori pikkus
- `dim()` – maatriksi mõõtmed
- Erinevad indeksi tüübid:
  - Täisarvud: `v[c(1,2,5)]`
  - Negatiivsed täisarvud: `v[-1]`
  - Loogiline indeks: `v[c(T, F, T)]`
  - Komponentide nimed: `v[c("beta", "gamma")]`
  - Kõik komponendid: `v[]`
- Vajalikele elementidele omistamine:
  `v[v < 0] <- 0`
- Andmebaasist selekteerimine:
  `data[data$income > 0,]`

Indekseerimine

Kuidas vektori/maatriksi vajalikku elementi näperdada

- `length()` – vektori pikkus
- `dim()` – maatriksi mõõtmed
- Erinevad indeksi tüübid:
  - Täisarvud: `v[c(1,2,5)]`
  - Negatiivsed täisarvud: `v[-1]`
  - Loogiline indeks: `v[c(T, F, T)]`
  - Komponentide nimed: `v[c("beta", "gamma")]`
  - Kõik komponendid: `v[]`
- Vajalikele elementidele omistamine:
  `v[v < 0] <- 0`
- Andmebaasist selekteerimine:
  `data[data$income > 0,]`

Arvutamine

- Põhilised matemaatilised operatsioonid: +, −, ∗, /
- Loogikatehted `!`, `&`, `|`, ==, <, <=, `%in%`, ...
- Täisarvuline jagamine \, jääk %%
- Maatrikskorrutis %*%
- Transponeerimine `t()`
- Igasugu (vektor)funktsioonid: `log()`, `sqrt()`, `exp()`, ...
- Vektorite operatsioonid: *recycling*
- Numbriline optimeerimine: `nlm()`, `optim()`
- Numbriline võrrandite lahendamine: `uniroot()`
- Numbriline integraal: `area()`
- `options(digits=)`

Arvutamine

- Põhilised matemaatilised operatsioonid: `+`, `-`, `*`, `/`
- Loogikatehted `!`, `&`, `|`, `==`, `<`, `<=`, `%in%`, ...
- Täisarvuline jagamine \, jääk %%
- Maatrikskorrutis %*%
- Transponeerimine `t()`
- Igasugu (vektor)funktsioonid: `log()`, `sqrt()`, `exp()`, ...
- Vektorite operatsioonid: *recycling*
- Numbriline optimeerimine: `nlm()`, `optim()`
- Numbriline võrrandite lahendamine: `uniroot()`
- Numbriline integraal: `area()`
- `options(digits=)`

Arvutamine

- Põhilised matemaatilised operatsioonid: +, −, ∗, /
- Loogikatehted `!`, `&`, `|`, ==, <, <=, `%in%`, ...
- Täisarvuline jagamine \, jääk %%
- Maatrikskorrutis %*%
- Transponeerimine `t()`
- Igasugu (vektor)funktsioonid: `log()`, `sqrt()`, `exp()`, ...
- Vektorite operatsioonid: *recycling*
- Numbriline optimeerimine: `nlm()`, `optim()`
- Numbriline võrrandite lahendamine: `uniroot()`
- Numbriline integraal: `area()`
- `options(digits=)`

Näide

- Laadi `http://www.obs.ee/~siim/ETU95.csv`
- Selekteeri vajalikud muutujad
- Arvuta vajalikud lähtesuurused
- Salvesta vahetulemused

Väljavõte ETU1995 andmebaasist

C18E0000	bruttopalk 1994 sügisel, EEK
G21	haridus: 1,2 – alg, 3,4 – kesk; 5-7 – kõrg
H01	perekonnaseis: 2,3 – (vaba)abielu
I01EKOOD	elukoha kood: 1 – Tallinn
J01	kas töötab uuringunädalal: 1 – jah, 2 – ei
L02A00	sünniaasta (kahekohaline)
L02D00	sugu: 1 – mees, 2 – naine

Näide

- Laadi `http://www.obs.ee/~siim/ETU95.csv`
- Selekteeri vajalikud muutujad
- Arvuta vajalikud lähtesuurused
- Salvesta vahetulemused

Väljavõte ETU1995 andmebaasist

C18E0000	bruttopalk 1994 sügisel, EEK
G21	haridus: 1,2 – alg, 3,4 – kesk; 5-7 – kõrg
H01	perekonnaseis: 2,3 – (vaba)abielu
I01EKOOD	elukoha kood: 1 – Tallinn
J01	kas töötab uuringunädalal: 1 – jah, 2 – ei
L02A00	sünniaasta (kahekohaline)
L02D00	sugu: 1 – mees, 2 – naine

Funktsioonid

- Interaktiivne käivitamine
- Argumendid
- Tulemused
- Kontrollstruktuurid:
  - `for()`
  - `break`
  - `if()`
  - `else`
- Trükkimine: `cat()`
- Silumine: `browser()`, `traceback()`
- Meetodid

Funktsioonid

- Interaktiivne käivitamine
- Argumendid
- Tulemused
- Kontrollstruktuurid:
  - `for()`
  - `break`
  - `if()`
  - `else`
- Trükkimine: `cat()`
- Silumine: `browser()`, `traceback()`
- Meetodid

Funktsioonid

- Interaktiivne käivitamine
- Argumendid
- Tulemused
- Kontrollstruktuurid:
  - `for()`
  - `break`
  - `if()`
  - `else`
- Trükkimine: `cat()`
- Silumine: `browser()`, `traceback()`
- Meetodid

Statistilised mudelid

- OLS – *linear model*

```
> model <- lm(response ~ explanatory + variables)
> summary(model)
```

- Logit/probit: osa üldistatud lineaarsetest mudelitest *generalised linear models*:

```
> model <- lm(response ~ explanatory + variables,
family=binomial(link="logit"))
> summary(model)
```

Statistilised mudelid

- OLS – *linear model*

```
> model <- lm(response ~ explanatory + variables)
> summary(model)
```

- Logit/probit: osa üldistatud lineaarsetest mudelitest *generalised linear models*:

```
> model <- lm(response ~ explanatory + variables,
family=binomial(link="logit"))
> summary(model)
```

Juhuslikud arvud

- Jaotused:

`.unif`	ühtlane jaotus
`.norm`	normaaljaotus
`.exp`	eksponentjaotus
`.chisq`	χ^2-jaotus
`.t`	t-jaotus
`.binom`	binoomjaotus
`.pois`	Poissoni jaotus
	...

- Statistilised tabelid:
  - `r...` juhuslike arvude generaator (`rnorm`)
  - `d...` tõenäosustihedus (`dnorm`)
  - `p...` kumulatiivne jaotusfunktsioon (`pnorm`)
  - `q...` jaotuse kvantiilid (`qnorm`)

Juhuslikud arvud

- Jaotused:
  - `.unif` ühtlane jaotus
  - `.norm` normaaljaotus
  - `.exp` eksponentjaotus
  - `.chisq` χ^2-jaotus
  - `.t` t-jaotus
  - `.binom` binoomjaotus
  - `.pois` Poissoni jaotus
  - ...
- Statistilised tabelid:
  - `r...` juhuslike arvude generaator (`rnorm`)
  - `d...` tõenäosustihedus (`dnorm`)
  - `p...` kumulatiivne jaotusfunktsioon (`pnorm`)
  - `q...` jaotuse kvantiilid (`qnorm`)

Maximum likelihood

ML tuleb teha nagu alati:

1. Kirjuta likelihoodi funktsioon
2. Maksimeeri parameetri järgi.

Näide: genereerime normaaljaotusega juhuslikke arve ja leiame valimi keskmise:

$$\ell_i(\mu, \sigma; x_i) = \log\left(\frac{1}{\sqrt{2\pi}\sigma}\exp\left(-\frac{1}{2}\frac{(x_i-\mu)^2}{\sigma^2}\right)\right) = \quad (1)$$

$$= -\frac{1}{2}\log(2\pi) - \log\sigma - \frac{1}{2}\frac{(x_i-\mu)^2}{\sigma^2} \quad (2)$$

Graafika

- Mõned näited
- Lihtne rida: `plot(x)`
- $x - y$ plot: `plot(x, y)`
- Histogramm: `hist(x)`
- Kernel tõenäosustihedus: `plot(density(x))`
- Võrdle jaotuse kvantiile: `qqnorm()`
- Funktsiooni kõver: `curve()`