Oulun yliopiston matemaattisten tieteiden tutkimusyksikkö/tilastotiede

805305A JOHDATUS REGRESSIO- JA VARIANSSIANALYYSIIN, sl 2022
(Jari Päkkilä)

Harjoitus 1

Tässä ensimmäisessä R-harjoituksessa kerrataan lyhyehkösti aiemmilla tilastotieteen kursseilla opittuja R-ympäristön toimintaperiaatteita. Aluksi on hyvä palauttaa mieleen pari käytännön seikkaa: (A) työhakemisto ja (B) R-skripti.

(A) Työhakemisto (*Working directory*). Kussakin erillisessä tilastollisessa analyysiprojektissa tarvittavat skripti- ja datatiedostot kannattaa sijoittaa tietokoneessa omaan hakemistoonsa. Tälle kurssille voit luoda tietokoneellesi oman alihakemiston esimerkiksi nimeltä `regvar`.

Kun sitten käynnistät R:n, niin ensimmäisenä tehtävänäsi on asettaa ao. projektille nimetty hakemisto istuntosi työhakemistoksi seuraamalla `RGui`-ikkunan vasemmasta ylänurkasta lähtien valikkopolkua `File - Change dir ...` Näin menetellen voit myös lukea istunnon aikana tarvitsemasi ulkoiset tiedostot ohjelman käytettäväksi kirjoittamalla lukukomennon pääargumentiksi pelkästään ao. tiedoston nimen. Jos tiedosto sijaitsee jossain muussa kansiossa, niin koko hakemistopolku pitää antaa lukukomennossa.

(B) R-skripti (*R script*) on ohjelmatiedosto, johon talletetaan ne R-komennot, jotka tarvitaan haluttujen analyysitehtävien suorittamiseksi. Skripti voidaan kirjoittaa esimerkiksi R:n omalla skriptieditorilla, jonka ikkuna aukeaa kun `RGui`-ikkunan vasemmasta ylänurkasta lähtien seurataan valikkopolkua `File - New script ...` Avoimeen ikkunaan voi alkaa kirjoittaa R-komentoja rivi kerrallaan. Rivien korjaus, muokkaaminen ja skriptin tallentaminen onnistuu `RGui`-ikkunan ylärivin valikkojen `File` ja `Edit` sisältämien tavallisten työkalujen avulla.

Kun kirjoitat R-skriptiin tarvittavat komentorivit, niin jätä rivien alusta pois allaolevissa tehtävissä näkyvät kehotemerkki (*prompt*) '>' sekä jatkorivin aloitusmerkki '+', jotka ilmestyvät konsoli-ikkunaan komentorivejä ajettaessa.

Yhden tai useammankin komentorivin aktivoiminen toteutetaan siten, että ensin nämä rivit maalataan, sitten kursori viedään `RGui`-ikkunan ylänurkassa olevan valikkorivin alapuolella olevan ikonirivin keskimmäisen ikonin ('*Run line or selection*') päälle ja lopuksi klikataan. Maalatut komentorivit siirtyvät konsoli-ikkunaan, ja ohjelma alkaa suorittaa niitä. – Vaihtoehtoisesti yhden komentorivin kerrallaan voi ajaa sijoittamalla kursori skriptin ao. riville ja painamalla `Ctrl-r`.

1. R:n käynnistäminen, työhakemiston käyttöönotto ja skriptin kirjoittaminen.

(a) Käynnistä R valitsemalla tietokoneen ohjelmavalikosta ja siirry työskentelemään R:n editori-ikkunalla.

(b) `RGui`-ikkunan valikosta `File` valitse `Change dir...`. Hae sitten tietokoneesi työhakemistoksi tälle kurssille määrittelemäsi alihakemisto, jos se sinulla on jo olemassa. Jos kurssikansiota ei vielä ole, niin voit nyt luoda sen käyttäen painiketta `Make New Folder`, jne.

(c) Aukaise R:n skriptieditori-ikkuna ja ala kirjoittaa siihen tämän istunnon R-komentoja. Voit kirjoittaa kommentteja merkin '#' jälkeen niin komentorivien väliin kuin myös rivien loppuun. Talleta skripti työhakemistoosi esimerkiksi nimellä `harj-1.R`.

2. Pienen aineiston sisäänluku, muokkaus ja alustava tarkastelu.

(a) R:n perusrakenne on vektori eli järjestetty jono, ja R käsitteleekin mm. havaintoaineiston muuttujia vektoreina. Talletetaan aluksi viiden kuvitteellisen miesopiskelijan pituudet vektoriin `pituus`. Tarvittavassa komennossa hyödynnetään R:n sijoitusoperaattoria "`<-`" ja funktiota `c()` (lyhennys sanasta "*concatenate*" = kytkeä peräkkäin), joka sitoo havaintoarvot vektoriksi:

```
> pituus <- c(183, 176, 173, 177, 185)
```

Talleta vastaavalla periaatteella kyseisten henkilöiden painot `paino`-nimiseen vektoriin. Painon havaintoarvot ovat 70, 76, 63, 69 ja 80.

(b) Vektorin alkiot voivat olla myös merkkijonoja, jolloin ne laitetaan lainausmerkkien sisään.

```
> tdk <- c("LuTK", "OyKKK", "TSTK", "LuTK", "OyKKK")
```

Vektorin sisältö voi olla myös loogista. Kokeile komentoja

```
> tdk == "OyKKK"
> sum(tdk == "LuTK")
> sum(pituus>175)
```

Moodle 1

(c) Tässä vaiheessa R:n muistiin on syötetty kolme erillistä vektoria; `pituus, paino` ja `tdk`. Muodostetaan näistä seuraavaksi pieni havaintomatriisi. Tähän tehtävään käytetään `data.frame()`-funktiota, joka sitoo muuttujat datakehikoksi. **Datakehikko** (*data frame*) on yksi R:n tärkeimpiä datarakenteista vektorin, matriisin ja listan ohella.

```
> opiskelijat <- data.frame(pituus, paino, tdk)
> opiskelijat
```

(d) Datakehikon rakennetta ja sen sisältämien muuttujien ominaisuuksia voi tutkia mm. funktiolla `str()`

```
> str(opiskelijat)
```

(e) Datakehikosta voidaan poimia tarvittassa yksittäinen rivi, sarake tai alkio. Tämä tapahtuu hakasulkeiden avulla seuraavasti:

```
> opiskelijat[2, ]    #   datakehikon 2. rivi
> opiskelijat[, 1]    #   datakehikon 1. sarake
> opiskelijat[4, 2]
```

Moodle 2

(f) Tietyn sarakkeen tai tietyn sarakkeen yksittäisen havainnon voi myös tulostaa käyttäen ao. saraketta vastaavan muuttujan nimeä erotettuna dollarilla datakehikon nimestä. Vertaa allaolevien rivien tuloksia edellisen kohdan vastaaviin.

```
> opiskelijat$pituus
> opiskelijat$paino[4]
```

(g) Kirjoita edellä luotu datakehikko levytiedostoon nimeltä `opiskelija.txt` omaan työhakemistoosi.

```
> write.table(opiskelijat, quote=FALSE, file = "opiskelija.txt")
```

(h) Katso sopivalla tekstinkäsittelyohjelmalla (esim. Notepad/muistio) miltä em. tiedosto näyttää.

3. Vektorit laskennassa.

(a) Lasketaan aineistomme opiskelijoille BMI-indeksin arvot kaavalla

$$\text{BMI} = \frac{\text{paino (kg)}}{\text{pituus}^2\ (\text{m}^2)}, \quad \text{jossa pituus on mitattu metreinä.}$$

Nykyisessä aineistossa vektori `pituus` sisältää pituuden arvot senttimetreissä. Muodostetaan ensin uusi vektori (muuttuja) `pituus.m` ja sen jälkeen haluttu vektori `bmi`.

```
> pituus.m <- pituus/100
> bmi <- paino/pituus.m^2
> bmi
```

Mikäli vektorit ovat samanmittaiset (kuten edellä, kaikissa vektoreissa viisi alkiota), kaikki matemaattiset operaatiot tehdään laskennassa alkioittain.

(b) Edellä BMI-arvot tulostuivat ruudulle usean desimaalin tarkkuudella. Tulostustarkkuutta voidaan säädellä tarvittaessa `round()`-funktiolla. Tulostetaan seuraavaksi BMI-arvot kahden desimaalin tarkkuudella.

```
> round(bmi,2)
```

Moodle 3

(c) Erilaisten tunnuslukujen laskennassa voidaan luonnollisesti hyödyntää valmiita R-funktioita. Esimerkiksi aritmeettinen keskiarvo voidaan laskea useilla eri tavoilla. Lasketaan tässä painon keskiarvo kolmella eri tavalla. Kokeillaan ensin keskiarvon määritelmän mukaista tapaa eli lasketaan painon havaintoarvot yhteen ja jaetaan saatu summa havaintojen lukumäärällä.

```
> sum(paino)/length(paino)
```

Yllä olevassa laskennassa `sum()`-funktio tekee havaintoarvojen yhteenlaskennan ja `length()` puolestaan laskee tarkasteltavan vektorin alkioiden lukumäärän. Keskiarvo voidaan laskea myös suoraan `mean()`-funktiolla tai vaikkapa `summary()`-funktion avulla, joka tulostaa samalla kertaa useampia tunnuslukuja.

```
> mean(paino)
> summary(paino)
```

(d) Muodostetaan seuraavaksi uusi vektori `painoja`, jossa on `paino`-vektorin havaintoarvojen lisäksi kaksi uutta havaintoa. Toinen näistä havainnoista on puuttuva tieto, jonka koodina R:ssä on NA (lyhenne termistä "*not available*").

```
> painoja <- c(paino, NA, 81)
> painoja
```

Kokeile laskea `painoja`-vektorin havaintoarvojen aritmeettinen keskiarvo edellä esitetyillä kolmella eri tavalla. Mitä huomaat? Onnistuiko laskenta kaikilla tavoilla?

Moodle 4

Joidenkin funktioiden yhteydessä puuttuvat tiedot aiheuttavat ongelmia laskennassa, ja siksi ne on poissuljettava argumentilla `na.rm=TRUE` (tulkinta: "*not available – remove*"). Esimerkiksi

```
> mean(painoja, na.rm=TRUE)
```

4. Kuvien piirtämisestä

R soveltuu hyvin erilaisten graafisten esitysten tekemiseen. Esimerkiksi tavanomaiset tilastokuviot voidaan yleensä piirtää varsin lyhyellä skriptinpätkällä valmiiden funktioiden avulla, ja kuvan yksityiskohtien muuttaminen oletusarvoistaan on mahdollista tiettyjen lisämääritysten avulla.

(a) Piirretään pituuden ja painon keskinäistä yhteyttä kuvaava sirontakuvio xy-koordinaatistoon. R piirtää halutun kuvan erilliseen grafiikkaikkunaan funktiolla `plot()`.

```
> plot(pituus, paino)
```

Komennolla saadaan aikaan funktion `plot()` oletusarvojen mukainen lopputulos. Erilaisissa kuvanpiirtotehtävissä käytettäville funktioille on olemassa tiettyjä kaikille graafisille funktioille yhteisiä lisäargumentteja, ns. **graafisia parametreja**, joista esitellään seuraavassa muutamia. Parannellaan edellä tehtyä kuvaa aluksi siten, että annetaan kuvalle otsikko parametrilla `main` ja nimetään samalla x- ja y-akselit parametreilla `xlab` ja `ylab`.

```
> plot(pituus, paino, main="Sirontakuvio", xlab="Pituus (cm)", ylab="Paino (kg)")
```

(b) Kuva-alue näyttää olevan tehokkaasti täytetty. Kummallakin akselilla kuva-alueen vaihteluväliä voisi kenties pidentää molempiin suuntiin. Tässä voidaan käyttää argumentteja `xlim` ja `ylim`, joilla saa annetuksi halutut vaihtelurajat.

Jos havaintojen symbolina käytetyn ympyrän haluaa vaihtaa, se tapahtuu parametrilla `pch`. Kokeilepa siis seuraavaksi lisätä komentoon `pch=15`. Mitä tämä sai aikaan? Moodle 5 Entä mitä muutoksia saavutetaan argumenttiasetuksilla `cex=2` ja `cex.lab=1.5`?

```
> plot(pituus, paino, main="Sirontakuvio", xlab="Pituus (cm)", ylab="Paino (kg)",
+    xlim=c(170, 190), ylim=c(60, 85), pch=15, cex=2, cex.lab=1.5)
```

(c) Kuvaan voidaan lisätä tarvittaessa myös vapaata tekstiä. Tämä tehdään funktiolla `text()`. Lyhimmässä muodossaan se tarvitsee kaksi määritystä: mihin kohtaan kirjoitetaan ja mitä kirjoitetaan. Sijaintitieto annetaan komennon alussa x- ja y-koordinaatteina. Kokeillaan lisätä punaisella kirjoitettua tekstiä alkaen koordinaattipisteestä (170, 80).

```
> plot(pituus, paino, main="Sirontakuvio", xlab="Pituus (cm)", ylab="Paino (kg)",
+    xlim=c(170, 190), ylim=c(60, 85), pch=15, cex=2, cex.lab=1.5)
> text(170, 80, "Kappas, tännehän voi kirjoittaa!", pos=4, col="red")
```

Edellä argumentti `pos=4` määrää sen, että teksti kirjoitetaan annetusta koordinaattipisteestä alkaen oikealle.

Siirrymme nyt analysoimaan isompaa havaintoaineistoa, joka on talletettu ulkoiseen tiedostoon. R-istunnossa tällaiset tiedostot luetaan sisään R:n datakehikoksi (*data frame*) joko komennolla `read.table()` tai jollakin muulla lukukomennolla riippuen luettavan tiedoston formaatista. Oletusarvona on tavanomainen vapaan formaatin ascii-tiedosto, joka on järjestetty havaintomatriisin muotoon siten, että sen rivit liittyvät eri havaintoyksiköihin, sarakkeet eri muuttujiin, ja sarakkeiden erottimena on välilyönti.

Laadi ja aja R-ohjelma suorittamaan seuraavia tehtäviä havaintoaineistosta, joka on kerätty The World Factbook -sivustolta (CIA:n julkaisema maantieteellinen vuosikirjasivusto osoitteessa `https://www.cia.gov/the-world-factbook/`). Analysoitavaan aineistoon on talletettu perustietoja maailman maista ja se on jaossa Moodlessa nimellä world.txt.

5. Tiedoston kopiointi omaan hakemistoon, sen sisäänluku R-ajovirtaan ja rakenteen tarkastelu

(a) Käy lataamassa tiedosto `world.txt` Moodlesta omalle koneellesi työhakemistoosi. Avaa tiedosto esim. *Notepad*-ohjelmalla tai vastaavalla ja tarkastele sen sisältöä.

(b) Kirjoita skriptiisi ja aja seuraavat komentorivit, joilla edellä mainittu datatiedosto luetaan R:n datakehikoksi `dake` ja tutki sen rakennetta

```
> dake <- read.table("world.txt", header=TRUE)
> str(dake)
```
Moodle 6

6. Luokkamuuttujien määrittely ja koodaus ja datakehikon kiinnittäminen ensisijaiseksi hakupolkuun.

(a) Muunna numeerisesti koodattu muuttuja `alue` R:n luokkamuuttujaksi eli **tekijäksi** (*factor*) ja anna selväkielisemmät nimet (`labels`) niiden tasoille:

```
> dake$ALUE <- factor(dake$alue,
+   labels = c("Australia", "Aasia", "Afrikka", "Eurooppa",
+              "P-Amerikka", "E-Amerikka") )
```

(b) "Kiinnitä" datakehikon muuttujat R:n ns. hakupolkuun ensisijaiseksi funktiolla `attach()` sekä tulosta ja tarkastele muuttujien suoria jakaumia

```
> attach(dake)
> summary(dake)
```

(c) Mitä tunnuslukuja esitetään eri muuttujista? Vertaa erityisesti muuttujien `alue` ja `ALUE` tunnuslukuja. Millainen on aineiston valtioiden aluejakauma? Montako Euroopan valtiota aineisto sisältää? Moodle 7 Kuinka monta puuttuvaa havaintoa on esimerkiksi elinajanodotteen (`elinodote`, vuosina) Moodle 8 ja nuorten työttömyysasteen (`nuortyot`, prosentteina) kohdalla?

Kun datakehikko `dake` on edellä "kiinnitetty", sen muuttujanimiin voi tästä lähtien viitata sellaisenaan. Jos kiinnittämistä ei tehdä, pitää jatkossa muuttujan edessä aina olla datakehikon nimi ja dollari, esim. `dake$alue` – ellei kutsuttavan tilastofunktion yhteydessä käytetä joko `data`-argumenttia tai `with`-komentoa. On myös muistettava, että tarpeelliset muuttujamuunnokset on syytä toteuttaa datakehikon sisällä (eli tyyliin `dake$ALUE <- factor(...)`) eikä datakehikosta irrallisille muuttujille (eli `ALUE <- ...`). – Yleisenä neuvona voi suositella komennon `attach()` säästeliästä käyttöä, koska datakehikoistaan irrallisena olevat muuttujat saattavat aiheuttaa sekaannuksia varsinkin, jos mukana on useita datakehikkoja joissa on kenties samannimisiä muuttujia.

7. Harjoitellaan seuraavaksi luokiteltujen muuttujien ristiintaulukointia ja prosenttijakaumien laskemista R-paketissa `Epi` olevan funktion `stat.table()` avulla.

(a) Lataa ao. paketti R-istunnossasi käytettäväksi ja tulosta muuttujien `ALUE` (sarakemuuttujaksi) ja `rannikko` (rivimuuttujaksi) välinen ristiintaulukko eli kontingenssitaulu:

```
> library(Epi)
>  stat.table( index = list( rannikko, ALUE ),
+      contents =  count(), data = dake)
```

Missä maanosissa ei ole lainkaan sellaisia valtioita, joissa ei ole lainkaan rannikkoa? Kuinka monta rannikkoa omaavaa valtiota Pohjois-Amerikassa on? Moodle 9

(b) Rivi- ja sarakemuuttujat voisi ehkä vaihtaa keskenään. Verrataan samalla eri maanosien rannikkomuuttujan prosenttijakaumia (`percent()`) keskenään maanosien välillä. Lisäksi kummankin muuttujan reunajakaumat (`margins`) olisivat informatiiviset. Edelleen, tiivistetään ristiintaulukossa `ALUE`en luokitusta siten, että Pohjois- ja Etelä-Amerikka kuuluvat samaan luokkaan. Tämä onnistuu `Epi`-paketin funktiolla `Relevel()`. Sijoita näin uudelleen luokitettu aluetieto muuttujaan `maanosa`. Tulosta uusi taulukko näiden ohjeiden mukaiseksi:

```
> dake$maanosa <- Relevel( dake$ALUE, list(1, 2, 3, 4, 5:6 ) )
> stat.table( index = list( maanosa, rannikko ),
+  contents = list( count(), percent(rannikko) ),
+  margins = TRUE, data = dake)
```

Miten tulkitset tulostusta? Monellako prosentilla aineiston Aasian maista on rannikkoa? Moodle 10 Entä mikä on vastaava prosenttiosuus kaikkien aineistoon kuuluvien valtioiden joukossa?

8. Elinajanodotteen (muuttuja `elinodote`, vuosina) jakauman alustava tarkastelu ja mahdollisten oudokkien eli muista selvästi poikkeavien arvojen identifiointi.

(a) Muodosta elinajanodotteen runko-lehtikuvio sekä laatikko-janakuvio. Käytä `with()` funktiota jolla määräät ohjelman ottamaan analysoitavan muuttujan datakehikosta `dake`. Tulosta samalla tämän muuttujan suoran jakauman perustunnusluvut.

```
> with(dake, stem(elinodote, scale=1.5))
> with(dake, boxplot(elinodote, horizontal=TRUE, xlab="Vuotta"))
> summary(elinodote)
> sd(elinodote, na.rm=TRUE)
```

Mitä havaintoja teet elinajanodotteen jakaumasta, sen keskimääräisistä arvoista ja hajonnasta? Moodle 11 Miten luonnehtisit jakauman muotoa? Onko havainnoissa oudokkeja eli outliereitä?

(b) Varsinkin pienillä aineistoilla käyttökelpoinen graafinen esitys on pistekuvio, jota piirtämään R:n perusgrafiikassa on tarjolla funktio `stripchart()`. Jos kuitenkin aineistossa on kovin lähekkäisiä ja/tai jopa päällekkäisiä arvoja, niin selkeämpi esitys saadaan paketin `beeswarm` funktiolla `beeswarm()`; suomeksi "mehiläisparvi". Lataa ao. paketti ja piirrä alekkain kummallakin em. funktiolla pistekuvio. Käytä tässä datakehikkoon sidottua muuttujaa pääargumenttina.

```
> library(beeswarm)
> par(mfrow=c(2,1))
> stripchart(dake$elinodote)
> beeswarm(dake$elinodote, horizontal=T, xlab="Vuotta")
```

(c) Edellisisssä kuvioissa näyttää olevan muutamia havaintoarvoja, joiden arvo on yli 85 vuotta. Kyseisten tilastoyksiköiden rivinumerot havaintoaineistossa on mahdollista tarkistaa funktiolla `which()` ja tulostaa sen jälkeen näytölle ehdon täyttävien valtioiden koko tietueet eli havaintoaineiston vastaavat rivit:

```
> which( elinodote > 85 )
> dake[ c(119, 157, 211) , ]
```

Mitkä valtiot täyttivät asetetun poimintaehdon? Moodle 12 Onko näiden maiden väestön kasvuaste (`vaestkasvu`, %) positiinen vai negatiivinen?

9. Uusien muuttujien luonti ja tarkastelu.

(a) Havaintoaineistoon on talletettu tiedot valtioiden pinta-aloista (`pintaala`, km^2) ja väestön määristä (`vaesto, lkm`). Näiden kahden perusmuuttujan avulla voidaan laskea valtioiden väestöntiheydet. Luo muuttuja `vaesttih` siten, että muodostettava muuttuja kertoo valtion väestön määrän neliökilometriä kohden. Luotava muuttuja voidaan samalla kiinnittää suoraan datakehikkoon, kun tälle sovelletaan funktiota `transform()`

```
> dake <- transform( dake, vaesttih = vaesto/pintaala)
> attach(dake)
```

(b) Piirrä väestöntiheyden jakaumasta laatikko-janakuvio funktiolla `boxplot()` ja tulosta myös jakauman tunnusluvut funktioilla `summary()` ja `sd()`

```
> par(mfrow=c(1,1))
> boxplot(vaesttih, horizontal=TRUE, xlab="Asukkaiden lkm neliökilometrillä")
> summary(vaesttih)
> sd(vaesttih, na.rm=TRUE)
```

Moodle 13

Piirretystä laatikko-janakuviosta nähdään, että kuvatussa muuttujassa on useita selkeitä outliereitä. Onko piirretty laatikko-janakuvio mielestäsi informatiivinen tälle jakaumalle? Millä kahdella maalla väestöntiheys on yli 15 000 asukasta/km^2. *Käytä ko. maiden nimien etsinnässä hyväksi kohdassa 8(c) esiteltyjä periaatteita.* Moodle 14

(c) Väestöntiheyden jakauma näyttäisi siis olevan vino ja se sisältää outliereitä. Kuvaisitko jakauman hajontaa mieluummin kvartiilivälin vai keskihajonnan avulla?

(d) Tulostetaan niiden alueiden asukasluvut ja pinta-alat, joiden väestöntiheys on yli 5000 asukasta neliökilometrillä.

```
> dake[which(vaesttih > 5000) , c("Maa", "pintaala", "vaesto")]
```

Ovatko tulostuneet alueet pinta-alaltaan suuria vai pieniä? Vertailun vuoksi todettakoon, että Suomen pinta-ala on noin 338 000 km^2.

10. Jatkuvan muuttujan jakauman kuvaaminen: histogrammi, silotettu histogrammi ja otoskertymäfunktion summakäyrä

(a) Kuvataan seuraavaksi elinajanodotteen jakaumaa histogrammin avulla.

```
> par(mfrow=c(1,1))
> hist(elinodote, main="Elinajanodotteen histogrammi")
```
Moodle 15

(b) Jos luokkarajoja ei erikseen anneta, R soveltaa oletusarvoisesti tiettyä kaavaa niiden määräämiseksi. histogrammin luokitusta voi muokata antamalla parametrin `breaks` arvoksi halutut luokkarajat. Kokeillaanpa vaihtelevanlevyisiä luokkavälejä seuraavasti.

```
> hist(elinodote, main="Elinajanodotteen histogrammi",
+     breaks=c(50, 60, 70, 75, 80, 85, 90) )
```

Vertaa oletusarvoilla piirrettyyn histogrammiin. Mitä havaintoja teet? Huomaa y-akselin erilainen mitta-asteikko.

(c) Jatkuvan muuttujan luokittelu on enemmän tai vähemmän väkivaltainen operaatio. Histogrammille vaihtoehtoinen esitys on silotettu (*smoothed*) histogrammi, joka antaa luonnollisemman estimaatin ao. muuttujan teoreettiselle tiheysfunktiolle. Tiheysfunktion estimoinnissa käyttökelpoinen funktio on `density()`. Piirrä sen avulla silotettu histogrammi elinajanodotteen jakaumalle koko aineistossa. **Huom!** Funktio `density()` toimii vain sellaiselle datavektorille, jossa ei ole puuttuvia havaintoja. Funktiota kutsuttaessa mukaan valitaan siis ao, muuttujan arvot sisältävästä vektorista vain ne havaintoyksiköt, joista elinajanodotteen arvo on olemassa, käyttäen valinnassa loogisen funktion `is.na()` (= "*is not available*" eli "on puuttuva") negaatiota (huutomerkki).

```
> plot( density(elinodote[!is.na(elinodote)]), lwd=2)
```

Vertaile histogrammia ja silotettua histogrammia keskenään piirtämällä molemmat alekkain samaan grafiikkaikkunaan.

```
> par(mfrow=c(2,1))
> hist(elinodote, main="Elinajanodotteen histogrammi")
> plot( density(elinodote[!is.na(elinodote)]), lwd=2)
```

11. Vertaillaan vielä vaihtoehtoisin graafisin esityksin elinajanodotteen jakaumia maanosien välillä.

(a) Aloitetaan vertailu laatikko-jana -kuvion avulla.

```
> par(mfrow=c(1,1))
> boxplot(elinodote ~ maanosa, horizontal=T, las=1)
```

Keskimäärin korkein elinajanodote näyttäisi olevan Euroopassa. Moodle 16 Moodle 17 Entäpä onko eroa vaihtelun määrässä?

(b) Jos vertailtavissa ryhmissä on kohtalaisen vähän havaintoja, jakauman/jakaumien kuvailuun voi käyttää myös pistekuviota. Jaetaan ennen seuraavien kuvien piirtoa `mfrow`-argumentilla kuvaikkuna neljään osaan (kaksi kuvariviä ja kaksi kuvasaraketta). Tarkkaile kuvia piirtäessäsi sitä, mitä `beeswarm()`-funktion lisäargumenteilla saadaan aikaan. Mikä näistä varianteista on oma suosikkisi?

```
> par(mfrow=c(2,2))
> beeswarm( elinodote ~ maanosa, cex=0.5)
> beeswarm( elinodote ~ maanosa, horizontal=T, method="center", , cex=0.5)
> beeswarm( elinodote ~ maanosa, horizontal=T, method="hex", pch=16, cex=0.5)
> beeswarm( elinodote ~ maanosa, horizontal=T, method="square", pch=16, cex=0.5)
```

Useamman histogrammin piirtäminen samaan kuvaan tuottaa tyypillisesti varsin epäselvän lopputuloksen. Selkeämpi ja havainnollisempi esitys saavutetaan piirtämällä ns. silotetut histogrammit erikseen vertailtaville ryhmille. Piirtämisen yhteydessä on hyvä rajata funktioilla `with()` ja `subset()` aineisto koskemaan vain tiettyä aluetta ja niitä havaintoyksikköjä, joilta kuvattavan muuttujan arvo on olemassa. Edellä piirretyissä kuvissa Pohjois- ja Etelä-Amerikka ovat olleet yhdistettynä samaan ryhmään. Piirretään seuraavaksi samaan kuvaan elinajanodotteen siloitetut histogrammit Pohjois- ja Etelä-Amerikan alueille

```
> par(mfrow=c(2,1))
> with( subset( dake, !is.na(elinodote) & ALUE=="P-Amerikka"),
+       plot(density(elinodote), lwd=2, col="blue", xlim=c(60,90) ) )
> with( subset( dake, !is.na(elinodote) & ALUE=="E-Amerikka"),
+       plot(density(elinodote), lwd=2, col="red", xlim=c(60,90) ) )
```

Miten kommentoisit jakaumien muotoja? Onko jakaumien sijainnissa eroa?

(c) Piirretään vielä Aasian, Afrikan ja Euroopan valtioiden bruttokansantuotteiden (`bkt`, BKT/asukas euroina) otoskertymäfunktiot samaan kuvaan. Ennen ko. kuvan piirtämistä kuvaikkuna palautetaan yhdeksi kokonaisuudeksi. Valituilla alueilla havaitut arvot vaihtelevat välillä 700–124 100 dollaria/asukas, joten Aasian otoskertymäfunktiota piirrettäessä varaudutaan argumentilla `xlim` siihen, että kaikki havainnot mahtuvat kuvaan mukaan.

```
> par(mfrow=c(1,1))
> plot(ecdf(bkt[ALUE=="Aasia"]), do.points=FALSE,
+    col = 'red', xlim=c(0,125000), verticals=T)
> plot(ecdf(bkt[ALUE=="Afrikka"]), do.points=FALSE,
+    col = 'green', add=TRUE, verticals=T)
> plot(ecdf(bkt[ALUE=="Eurooppa"]), do.points=FALSE,
+    col = 'blue', add=TRUE, verticals=T)
# nimetään käyrät
> legend(100000, 0.3, legend=c("Aasia", "Afrikka", "Eurooppa"),
+     lty=c(1,1,1), col=c("red","green","blue"))
```

Miltä tasolta alkaen Aasian ja Euroopan otoskertymäfunktiot alkavat kulkea (likimain) yhteneväisesti? Moodle 18 Arvioi edellä piirtämäsi kuvion avulla bruttokansantuotteen mediaani ja kvartiilit Euroopan maiden joukossa.

(d) Lopuksi voimme tulostaa ja vertailla bruttokansantuotteen jakauman perustunnuslukuja eri maanosissa

```
> with(dake, tapply(bkt, ALUE, summary))
```

(e) Tarkista aluksi tulostuksesta edellisessä kohdassa tekemäsi arviot Euroopan maiden bruttokansantuotteen mediaanista ja kvartiileista. Mitä päätelmiä teet bruttokansantuotteen eroista eri alueiden välillä? Moodle 19 Moodle 20

12. Muokatun datakehikon tallettaminen levytiedostoksi. Kirjoita uusi datakehikko omaan hakemistoosi tiedostonimellä `world2.txt`

```
> write.table(dake,file = "world2.txt", quote=F)
```