StatData 2 - Case 3: Brug af en baselinemåling

Anders Tolver

20 Sep 2017

Ugens case handler om brug af en baseline-måling, altsa en måling på hver forsøgsenhed af samme type som responsen blot taget før behandlingen af forsøgsenheden. Opgaven skal ved hjælp af et eksempel illustrere fordelen ved at benytte den metode, der anbefales i kompendiet (BMS).

Beskrivelse af data

For at undersøge effekten af 7 forskellige behandlinger (hver behandling var et bestemt måltid) deltog 10 personer 7 gange hver, så hver person fik prøvet alle 7 måltider. Forsøget var inddelt i 7 perioder, og i hver periode prøvede hver person et af måltiderne, men rækkefølgen af måltiderne var randomiseret for hver person. Efter at personen havde spist måltidet, blev han/hun bedt om at markere sin fornemmelse af sult på en skala fra 0 til 100. Dette skete et antal gange med fastlagte mellemrum, og i datasættet som vi ser på her angiver variablen `appetite` gennemsnitsscoren for den pågældende person ved den pågældende behandling. En tilsvarende registrering af personens sult-fornemmelse blev foretaget lige inden måltidet; denne måling er kaldt `baseline` i datasættet. En enkelt person deltog ikke i en enkelt af perioderne, så datasttet indeholder 69 forsøgsenheder i alt (altså 10 personer gange 7 måltider med et enkelt bortfald).

Link til datasættet `caseuge3.txt` kan findes via kursusoversigten under ugeplanen for uge 3.

Undersøgelse af datasættet

a. Gem filen `caseuge3.txt` med data, opret et R markdown dokument som gemmes i samme mappe som data, og indsæt en *code chunk* i dit datasæt, der indlæser data fra filen `caseuge3.txt`. Gem datasættet i R under navnet `case3`. Find ud af hvilke variable datasættet indeholder.

b. Kør følgende to R-kommandoer og diskuter, hvad det fortæller dig om forsøgsplanen. Er forsøget balanceret i de faktorer der indgår i forsøgsdesignet?

```
table(case3$treat)
table(case3$treat, case3$period)
```

Statistiske modeller

c) Responsvariablen i datasættet hedder `appetite`. Opstil (på papir) en statistisk model hvori variablene `period`, `person` og `treat` alle indgår. Hvilke af disse tre variable bør indgå i modellen som faktorer, og hvilke niveauer har hver af faktorerne? (Undlad både i dette og alle følgende sprøgsmål at inkludere vekselvirkninger i modellen).

d) Estimer parametrene i din model i R, og test hypotesen om at der ikke er effekt af behandlingerne. Noter (på papir) F-teststørrelsen og P-værdien for testet samt estimatet for residualspredningen (s).

Det er nærliggende at forestille sig at personerne måske ikke føler sig lige sultne når de ankommer (før måltidet), og at det også vil påvirke deres appetit efter måltidet. Baselinemålingen giver information om sult-fornemmelsen før måltidet, men udfordringen ligger i, hvordan vi bedst bruger denne information ved den statistiske analyse. En ofte benyttet metode er at analysere måltidets effekt på ændringen i sult-fornemmelsen fra før til efter måltidet. Man vil med andre ord trække baseline målingen fra scoren for sult-fornemmelse der måles efter måltidet. Ofte vil man sige, at vi benytter ændringen i sult-fornemmelsen som responsvariable ved den statistiske analyse.

e) Opskriv (på papir) modellen svarende til c., nu blot med ændringen i sult-fornemmelse fra før til efter måltidet som responsvariabel. Estimer modellen i R og test igen hypotesen om at der ikke er effekt af behandlingerne, og noter igen F-teststørrelse, P-værdi og residualspredning. Det kan være nyttigt at lave en ny variable ved brug af følgende R-kode

```
case3$change <- case3$appetite - case3$baseline
```

f) Prøv at drage en (foreløbig) konklusion ved at sammenligne resultaterne af testene fra spørgsmål d. og e.?

Som sidste mulighed lader vi nu variablen `baseline` indgå i modellen som kovariat. Vi vender i den forbindelse tilbage til at bruge `appetite` som responsvariabel (i modsætning til i delspørgsmål e., hvor vi benyttede ændringen i forhold til baseline).

g) Opskriv (på papir) modellen fra spørgsmål c., nu blot med den forskel at `baseline` indgar som kovariat. Estimer modellen i R og test igen hypotesen om at der ikke er effekt af behandlingerne. Husk at notere F-teststørrelse, P-værdi og residualspredning.

h) Sammenlign resultaterne fra g. med dem fra d. og f. Hvilken af de to forrige minder resultaterne mest om? Fokuser f.x. på sammenligning af residualspredningen eller overordnede konklusioner vedrørende behandlingens effekt på sult-fornemmelsen.

i) Estimer koeffcienten (=hældningen) hørende til kovariaten (`baseline`) i modellen fra g. Opskriv de tre modeller fra c., e. og g. (på papir!). Hvilke værdier skulle koefficienten fra modellen i g. antage for, at modellen (dvs. ligningen for den statistiske model) svarer til modellerne fra hhv. c. og e.?

Estimater og modelkontrol

Hvis der er mere tid, kan du tage fat på at gøre analysen af data helt færdig, for det er den ikke med ovenstående, idet vi mangler modelkontrol samt estimater og konfidensintervaller for interessante effekter (her: behandlingerne).

j) Tag udgangspunkt i modellen fra delspørgsmål g. Lav et residualplot og et QQ-plot for modellen og kommenter.

k) Overvej, hvordan man kan kontrollere antagelserne bag modellen, der blandt andet siger at der er en retlinet sammenhæng mellem `baseline` og `appetite`. Søg inspiriation i den del af kursusmaterialet, der diskuterer muligheden for at lade `baseline` indgå som en kvadratisk effekt.

Bliv bedre til R

På Statistisk Dataanalyse 2 er det primære fokus rettet mod, at I lærer at lave statistiske modeller i R herunder at I kan aflæse og fortolke output. De fleste datasæt bliver altid serveret i et ensartet format, og det er sjældent en væsentlig del af forelæsninger og opgaver, at I selv kan lave datamanipulation og figurer. Det er oplagt, at hvis man selv vil arbejde professionelt med R efter kurset, så opstår der hurtigt et behov for at kunne lave helt basale datamanipulationer og grafik. Ved at grave lidt i R programmerne fra forelæsningerne, kan man lære en del i den retning.

Hvis du har tid og lyst, kan du bruge de følgende afsnit, til at snuse til nogle smarte funktioner i R, der kan være nyttige, når/hvis du senere skal arbejde med dine egne data i R.

Moderne værktøjer til datamanipulation

Der findes en samling af R pakker, der forsøger at gøre det lettere at udføre forskellige operationer på datasæt, som man ofte vil have behov for, hvis man arbejder med et konkret datasæt.

Denne sampling af R pakker kan installeres ved at installere R pakken som hedder `tidyverse` (i praksis installeres herved en hel sampling af pakker). Dernæst kan du som altid loade pakken i dit R markdown dokument ved at skrive

```
library(tidyverse)
```

Der findes en række nye og mere effektive funktioner til indlæsning af data, der alle har et navn af formen `read_`. Fx. findes en funktion `read_table2()`, der kan bruges til at indlæse data, og som fungerer stort set lige som `read.table()` .

```
case3new <- read_table2("../data/caseuge3.txt")
```

```
## Parsed with column specification:
## cols(
##   person = col_integer(),
##   period = col_integer(),
##   treat = col_integer(),
##   baseline = col_integer(),
##   appetite = col_double()
## )
```

```
case3new
```

```
## # A tibble: 69 x 5
##    person period treat baseline appetite
##     <int>  <int> <int>    <int>    <dbl>
##  1      1      1     4       49     58.8
##  2      1      3     1       44     39.3
##  3      1      4     3       55     60.5
##  4      1      5     2       41     57.4
##  5      1      6     5       42     62.9
##  6      1      7     7       48     62.0
##  7      2      1     3       41     61.5
##  8      2      2     5       48     43.5
##  9      2      3     7       59     48.9
## 10      2      4     4       67     48.6
## # ... with 59 more rows
```

Funktionen `mutate()` kan anvendes på et datasæt (her på `case3new`) til at lave nye variable (og til at lave eksisterende variable om til faktorer)

```
case3new <- mutate(case3new, person = factor(person), treat = factor(treat),
                   period_fac = factor(period), change = appetite - baseline)
```

```
## Warning: package 'bindrcpp' was built under R version 3.3.2
```

```
case3new
```

```
## # A tibble: 69 x 7
##    person period  treat baseline appetite period_fac change
##    <fctr>  <int> <fctr>    <int>    <dbl>     <fctr>  <dbl>
##  1      1      1      4       49     58.8          1    9.8
##  2      1      3      1       44     39.3          3   -4.7
##  3      1      4      3       55     60.5          4    5.5
##  4      1      5      2       41     57.4          5   16.4
##  5      1      6      5       42     62.9          6   20.9
##  6      1      7      7       48     62.0          7   14.0
##  7      2      1      3       41     61.5          1   20.5
##  8      2      2      5       48     43.5          2   -4.5
##  9      2      3      7       59     48.9          3  -10.1
## 10      2      4      4       67     48.6          4  -18.4
## # ... with 59 more rows
```

Funktionen `count()` kan bruges til at lave tabeller for udvalgte variable i et datasæt.

```
count(case3new, treat)
```

```
## # A tibble: 7 x 2
##    treat     n
##   <fctr> <int>
## 1      1    10
## 2      2    10
## 3      3    10
## 4      4    10
## 5      5    10
## 6      6     9
## 7      7    10
```

```
count(case3new, period)
```

```
## # A tibble: 7 x 2
##   period     n
##    <int> <int>
## 1      1    10
## 2      2     9
## 3      3    10
## 4      4    10
## 5      5    10
## 6      6    10
## 7      7    10
```

```
count(case3new, treat, period)
```

```
## # A tibble: 40 x 3
##     treat period     n
##    <fctr>  <int> <int>
##  1      1      1     3
##  2      1      3     1
```

```
##  3      1      4     1
##  4      1      5     1
##  5      1      7     4
##  6      2      2     2
##  7      2      4     2
##  8      2      5     3
##  9      2      6     3
## 10      3      1     2
## # ... with 30 more rows
```

Man kan lave et nyt datasæt, men udvalgte observationer ved brug af funktionen `filter()`. Her udtrække f.x. data fra person nummer 4

```
data_pers4 <- filter(case3new, person == 4)
data_pers4
```

```
## # A tibble: 7 x 7
##   person period  treat baseline appetite period_fac change
##   <fctr>  <int> <fctr>    <int>    <dbl>     <fctr>  <dbl>
## 1      4      1      1       35     65.4          1   30.4
## 2      4      2      4       55     77.6          2   22.6
## 3      4      3      6       50     50.1          3    0.1
## 4      4      4      5       52     76.5          4   24.5
## 5      4      5      2       62     67.9          5    5.9
## 6      4      6      3       58     79.6          6   21.6
## 7      4      7      7       63     75.7          7   12.7
```

Funktionen `select()` bruges til at udvælge et deldatasæt med kun udvalgte søjler (dvs. variable). Her laves et datasæt, som indholder søjle 1 til 5 og et datasæt, der kun indeholder variablene `treat, appetite, change`.

```
data_1_5 <- select(case3new, 1:5)
data_1_5
```

```
## # A tibble: 69 x 5
##    person period  treat baseline appetite
##    <fctr>  <int> <fctr>    <int>    <dbl>
##  1      1      1      4       49     58.8
##  2      1      3      1       44     39.3
##  3      1      4      3       55     60.5
##  4      1      5      2       41     57.4
##  5      1      6      5       42     62.9
##  6      1      7      7       48     62.0
##  7      2      1      3       41     61.5
##  8      2      2      5       48     43.5
##  9      2      3      7       59     48.9
## 10      2      4      4       67     48.6
## # ... with 59 more rows
```

```
data_small <- select(case3new, treat, appetite, change)
data_small
```

```
## # A tibble: 69 x 3
##     treat appetite change
##    <fctr>    <dbl>  <dbl>
##  1      4     58.8    9.8
##  2      1     39.3   -4.7
```

```
##  3      3     60.5    5.5
##  4      2     57.4   16.4
##  5      5     62.9   20.9
##  6      7     62.0   14.0
##  7      3     61.5   20.5
##  8      5     43.5   -4.5
##  9      7     48.9  -10.1
## 10      4     48.6  -18.4
## # ... with 59 more rows
```

Nogle af de mere nyttige tricks består i at kunne kombinere forskellige kommandoer fra `tidyr udvalgte størrelser, men hvor udregningerne foretages separat for hvert niveau af faktoren`treat'.

```
summarise(group_by(case3new, treat), mean_app = mean(appetite)
          , sd_app = sd(appetite), mean_change = mean(change) )
```

```
## # A tibble: 7 x 4
##    treat mean_app   sd_app mean_change
##   <fctr>    <dbl>    <dbl>       <dbl>
## 1      1 61.84000 19.22008   10.440000
## 2      2 64.66000 17.22138    5.360000
## 3      3 67.11000 18.12778    8.410000
## 4      4 65.15000 19.74123    9.150000
## 5      5 66.84000 14.17746    7.840000
## 6      6 59.53333 15.38075   -5.911111
## 7      7 61.88000 19.52160    4.580000
```

Pæne figurer i R

Installer R pakken `ggplot2` på din computer og sørg for, at du har indlæst datasættet til casen og navngivet det `case3new` i R.

Følgende R kode kan bruges til at lave forskellige scatterplot, hvor variablene `baseline` og `appetite` plottes mod hinanden. Prøv at køre de forskellige R kommandoer for at finde ud af, hvad de forskellige dele af koden gør.

```
library(ggplot2)
ggplot(data = case3new) + geom_point(aes(x = baseline, y = appetite))
ggplot(data = case3new) + geom_point(aes(x = baseline, y = appetite)
                                     , color = "blue", size = 3, shape = "x")
ggplot(data = case3new) + geom_point(aes(x = baseline, y = appetite
                                         , color = person, size = period))
ggplot(data = case3new) +
  geom_point(aes(x = baseline, y = appetite, color = person, size = period)) +
  labs(x = "Sult-fornemmelse ved baseline", y = "Sult-fornemmelse efter maaltid"
       , title = "Data fra case 3", subtitle = "21/9-2017")
```