Statistisk Dataanalyse 2 — 27. februar 2013
Institut for Matematiske Fag — Kasper Brink

Afleveringsopgave 2

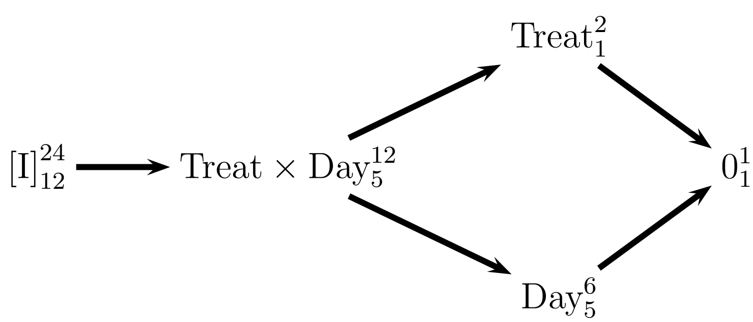
1

Starter med at indlæse data

```
afl2data = read.table("afl2.txt", header = T)
head(afl2data)

##   treat day logco2
## 1   raj   2   0.42
## 2   raj   2   0.37
## 3   raj   4   1.02
## 4   raj   4   1.27
## 5   raj   6   1.64
## 6   raj   6   1.46
```

Det ses at datasættet indeholder 3 variable, responsen `logco2` samt de forklarende `treat` og `day`. `treat` kan kun opfattes som en faktor, mens `day` kan opfattes både som faktor og numerisk variabel. Der er gentagelser af kombinationerne af `treat` og `day` så det er muligt at starte med en model med vekselvirkning mellem de to.

2



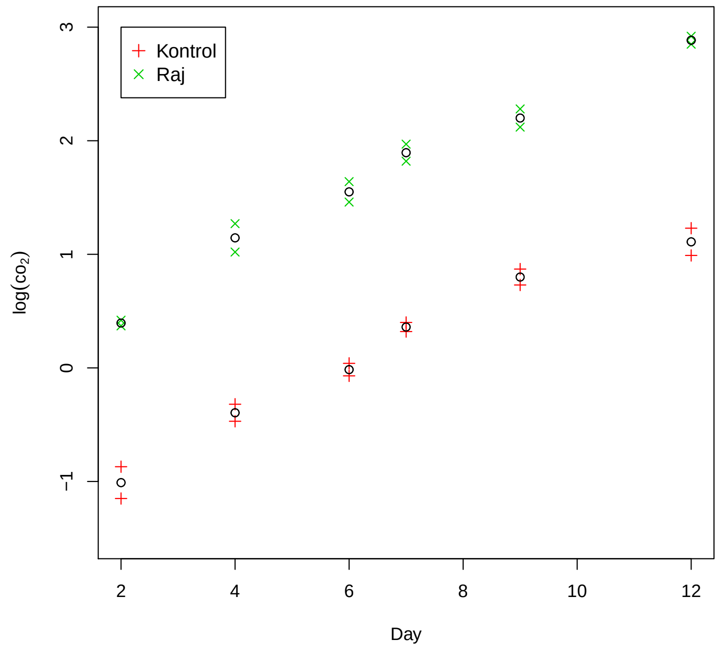
Figur 1: Faktordiagram til den fulde model med vekselvirkning

Model1: $Y_i = \gamma(Treat \times Day_i) + e_i$, hvor $e_1, \ldots, e_{24}$ er iid $\sim N(0, \sigma^2)$
Fittes i `R`

```
model1 <- lm(logco2 ~ treat * factor(day))
```

3

Den fulde model kan reduceres til den additive model, der ikke kan reduceres ydeligere, som det ses i Tabel 1
Model2: $Y_i = \alpha(Treat_i) + \beta(Day_i) + e_i$, hvor $e_1, \ldots, e_{24}$ er iid $\sim N(0, \sigma^2)$ Fittes i `R`



Figur 2: Illustration af den fulde model med vekselvirkning. Hver kombination af niveauerne for de to faktorer tilpasses en parameter. Dermed kan effekten af den ene behandling afhænge af niveauet af den anden. Fittede værdier er vist med cirkler

```
model2 <- lm(logco2 ~ treat + factor(day))
```

Tabel 1: Model og test skema

Model	Factors	Mean	SS_e	DF_e	Test	Test of factor	F	p-value
1	$T \times D$	$\gamma(T \times D_i)$	0.1735	12	–	–	–	–
2	$T + D$	$\alpha(T_i) + \beta(D_i)$	0.2671	17	2 vs 1	$T \times D$	1.2952	0.3287
3	T	$\alpha(T_i)$	13.7347	22	3 vs 2	D	171.4119	< 0.0001
4	D	$\beta(D_i)$	14.4352	18	4 vs 2	T	901.6364	< 0.0001

LSD i den additive model

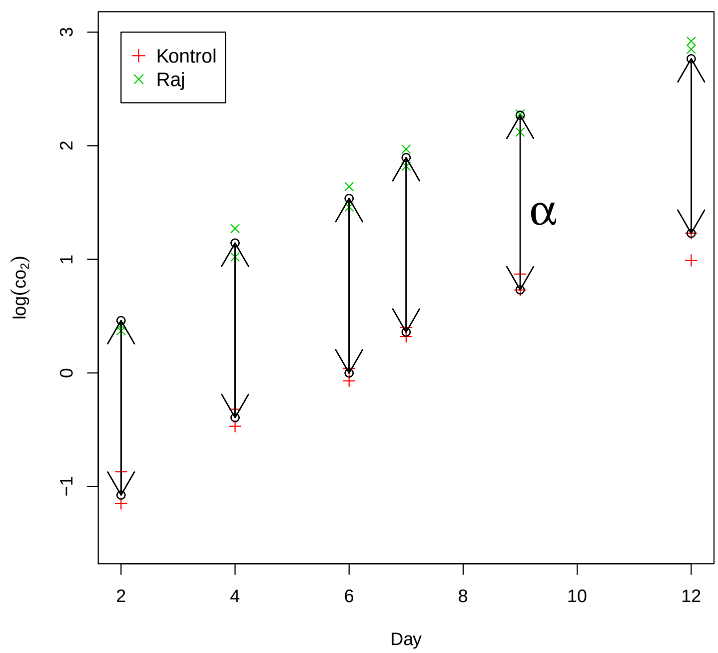
$\text{LSD}_{Day} \qquad t_{0.975,df} \times s \times \sqrt{\frac{2}{k \times n}}$

$\text{LSD}_{Day} \qquad 2.1098 \times 0.1254 \times \sqrt{\frac{2}{2 \times 2}} = 0.187$

Der kan naturligvis også beregnes en LSD_{Treat}, men den giver ikke ny information. Da der kun er to niveauer af `treat` og da hovedvirkningen ikke kan udelades fra modellen, må de nødvendigvis være signifikant forskellige.

Estimater i den additive model

Der er mange må der at få estimaterne fra en additiv model ud på i `R`. De bør alle give samme resultat, men estimaterne skal kombineres forskelligt. Her følger et par eksempler.



Figur 3: Illustration af den additive model. I denne model er effekten af den ene behandling ens for alle niveauer af den anden behandling. Effekten af treat er altså den samme for alle dage. Størrelsen af effekten er α som vil kunne findes i summary som treatraj 1.5367

```
summary(model2)$coef

##                Estimate Std. Error t value  Pr(>|t|)
## (Intercept)     -1.0758    0.06770  -15.89 1.235e-11
## treatraj         1.5367    0.05118   30.03 3.598e-16
## factor(day)4     0.6825    0.08864    7.70 6.120e-07
## factor(day)6     1.0750    0.08864   12.13 8.555e-10
## factor(day)7     1.4350    0.08864   16.19 9.172e-12
## factor(day)9     1.8075    0.08864   20.39 2.179e-13
## factor(day)12    2.3050    0.08864   26.00 3.949e-15
```

Her skal man selv regne 'alt' ud. `(Intercept)` er kontrolbehandlingen til dag 2, `factor(day)4,6,7,9,12` er forskellen på den givne dag og dag 2. `treatraj` er forskellen på kontrol og rajgræs der jo er ens for alle dage.

```
model2a <- lm(logco2 ~ factor(day) + treat - 1)
summary(model2a)$coef

##                  Estimate Std. Error   t value  Pr(>|t|)
## factor(day)2  -1.0758333    0.06770 -15.89139 1.235e-11
## factor(day)4  -0.3933333    0.06770  -5.81002 2.090e-05
## factor(day)6  -0.0008333    0.06770  -0.01231 9.903e-01
## factor(day)7   0.3591667    0.06770   5.30534 5.814e-05
## factor(day)9   0.7316667    0.06770  10.80763 4.902e-09
## factor(day)12  1.2291667    0.06770  18.15632 1.445e-12
## treatraj       1.5366667    0.05118  30.02726 3.598e-16
```

Her er forskellene på dage udregnet, man kan direkte aflæse estimater for kontrol behandling til alle dage. For at få estimaterne til raj lægges `treatraj` til.

```
model2b <- lm(logco2 ~ treat + factor(day) - 1)
summary(model2b)$coef

##                Estimate Std. Error t value  Pr(>|t|)
## treatkontrol   -1.0758    0.06770 -15.891 1.235e-11
## treatraj        0.4608    0.06770   6.807 3.054e-06
## factor(day)4    0.6825    0.08864   7.700 6.120e-07
## factor(day)6    1.0750    0.08864  12.128 8.555e-10
## factor(day)7    1.4350    0.08864  16.189 9.172e-12
## factor(day)9    1.8075    0.08864  20.392 2.179e-13
## factor(day)12   2.3050    0.08864  26.004 3.949e-15
```

Ved at bytte om på `treat` og `factor(day)` bliver forskellen mellem kontrol og raj til dag 2 udregnet, man skal så selv udregne for de øvrige dage.
Konklusionen er at det godt kan betale sig at overveje den rækkefølge man har på faktorerne i sin model. Hvis der er en faktor med mange niveauer er det praktisk at have den først så man kan få udregnet estimaterne ved at skrive `-1` i modellen. Hvis den anden faktor som i dette tilfælde har få niveauer er det nemt selv at lægge estimaterne til.

4

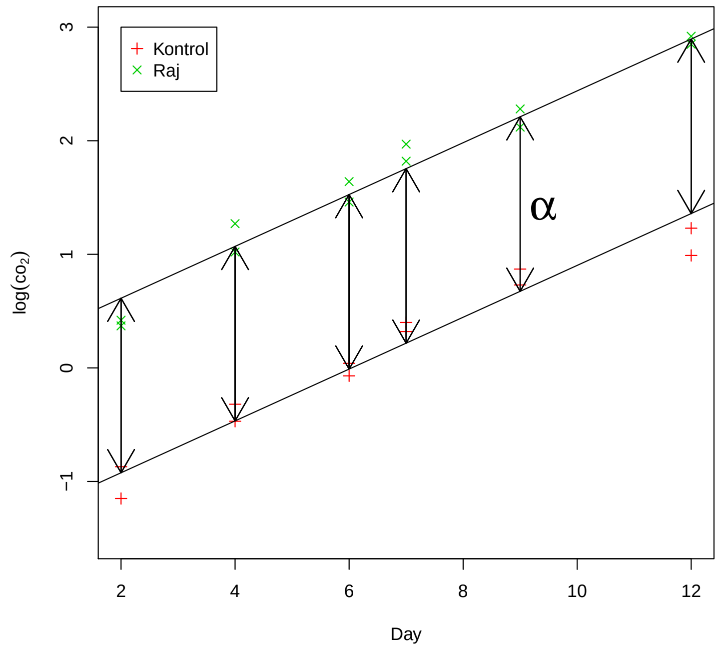
Opstiller en model med lineær sammenhæng mellem `logco2` og `day`. Der tilpasses en linie for hvert niveau af `treat` som det ses i Figur 4.
Model5: $Y_i = \alpha(Treat_i) + \beta * Day_i + e_i$, hvor $e_1, \ldots, e_{24}$ er iid $\sim N(0, \sigma^2)$
Fittes i `R` og testes mod den additive model det ses at reduktionen ikke er mulig

```
model5 <- lm(logco2 ~ treat + day)
anova(model5, model2)

## Analysis of Variance Table
##
## Model 1: logco2 ~ treat + day
## Model 2: logco2 ~ treat + factor(day)
##   Res.Df   RSS Df Sum of Sq    F Pr(>F)
## 1     21 0.543
## 2     17 0.267  4     0.276 4.38  0.013
```

Parametrene findes i summary på følgende måde:

```
summary(model5)$coef

##              Estimate Std. Error t value  Pr(>|t|)
## (Intercept)  -1.3796    0.08177  -16.87 1.092e-13
## treatraj      1.5367    0.06563   23.42 1.568e-16
## day           0.2282    0.01010   22.59 3.228e-16
```

- `(Intercept)` er værdien af $\log(CO_2)$ ved `day`=0 for kontrol behandlingen.
- `treatraj` $= \alpha$ er effekten af behandling med raj i forhold til kontrolbehandlingen.
- `day`$=\beta=$ ændringen i $\log(CO_2)$ ved ændring i `day` på en enhed.



Figur 4: Illustration af den retliniede model. Der tilpasset en ret linie som funktion af day til hvert niveau af treat. Linierne er parallelle med afstand α som kan findes i summary under `treatraj`

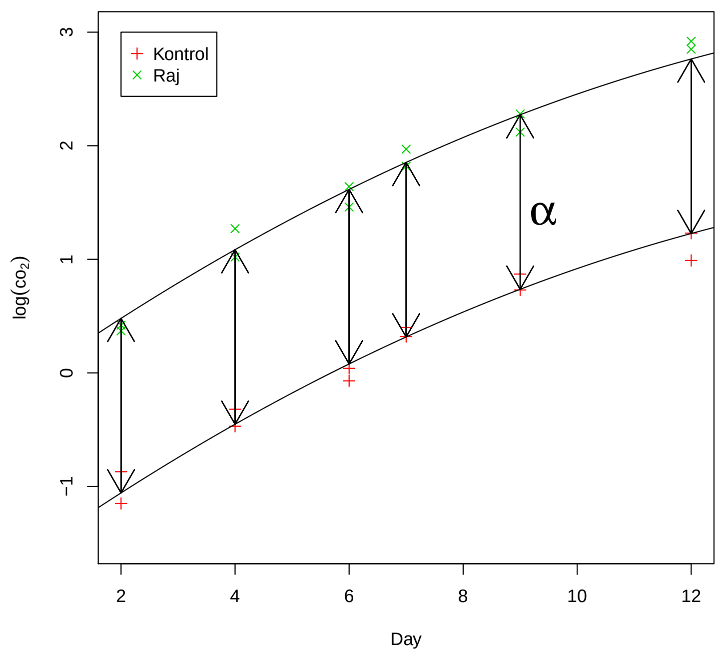
6

Som alternativ model forsøges der med en model med et kvadratled.
Model6: $Y_i = \alpha(Treat_i) + \beta_1 * Day_i + \beta_2 * Day_i^2 + e_i$, hvor $e_1, \ldots, e_{24}$ er iid $\sim N(0, \sigma^2)$
Fittes i `R` og testes mod den adiitive model. Reduktion til model med kvadratled er mulig.

```
day2 = day * day
model6 = lm(logco2 ~ treat + day + day2)
anova(model6, model2)

## Analysis of Variance Table
##
## Model 1: logco2 ~ treat + day + day2
## Model 2: logco2 ~ treat + factor(day)
##   Res.Df   RSS Df Sum of Sq    F Pr(>F)
## 1     20 0.315
## 2     17 0.267  3     0.048 1.02   0.41
```

Parametrene findes i summary på følgende måde:

```
summary(model6)$coef

##              Estimate Std. Error t value  Pr(>|t|)
## (Intercept) -1.735397   0.113317  -15.31 1.640e-12
## treatraj     1.536667   0.051244   29.99 4.230e-18
## day          0.357843   0.035016   10.22 2.193e-09
## day2        -0.009247   0.002433   -3.80 1.122e-03
```



Figur 5: Illustration af den kvadratiske model. Der er tilpasset en parabel som funktion af day til hvert niveau af treat. Linierne er parallelle med afstand α, som findes i summary under `treatraj`

- `(Intercept)` er værdien af $\log(CO_2)$ ved `day`$=0$ for kontrol behandlingen.
- `treatraj` $= \alpha$ er effekten af behandling med rajgræs i forhold til kontrolbehandlingen.
- `day`$=\beta_1$ `day2`$=\beta_2$

7

Det er en god øvelse at regne estimatet i hånden, det kan gøres på følgende måde:
Intecept + effekt af behandling med rajgræs $+\beta_1 *$day$+\beta_2 *$day^2
Ved indsættelse fås

```
(est1 = -1.7354 + 1.53667 + 0.3578 * 10 - 0.009247 * 100)

## [1] 2.455
```

For at få konfidensintervallet bruges `estimable` i pakken gmodels. Først laves den ønskede lineære kombination af parametrene, den skal svare til det man ville gøre med håndkraft. Derefter beder man `estimable` om at bruge denne kombination på parametrene i modellen samt at beregne et konfidensinterval på 95% niveau.

```
est2 = c(1, 1, 10, 100)
library(gmodels)
estimable(model6, est2, conf.int = 0.95)

##                Estimate Std. Error t value DF Pr(>|t|) Lower.CI Upper.CI
## (1 1 10 100)    2.455    0.04497    54.6 20        0    2.361    2.549
```