Opgavebesvarelse, korrelerede målinger

I 18 familier bestående af far, mor og 3 børn (i veldefinerede aldersintervaller, med `child1` som det ældste barn og `child3` som det yngste) har man registreret antallet af infektioner over et bestemt tidsinterval.

Familierne er inddelt efter, hvor tæt, de bor sammen, angivet ved faktoren *crowding*, som har 3 niveauer, `uncrowded`, `crowded` og `overcrowded`.

Data ses i tabellen nedenfor:

Table 8.13. Numbers of swabs positive for pneumococcus during fixed periods

Crowding category	Family serial number	Family status					
				Child			
		Father	Mother	1	2	3	Total
Overcrowded	1	5	7	6	25	19	62
	2	11	8	11	33	35	98
	3	3	12	19	6	21	61
	4	3	19	12	17	17	68
	5	10	9	15	11	17	62
	6	9	0	6	9	5	29
		41	55	69	101	114	380
Crowded	7	11	7	7	15	13	53
	8	10	5	8	13	17	53
	9	5	4	3	18	10	40
	10	1	9	4	16	8	38
	11	5	5	10	16	20	56
	12	7	3	13	17	18	58
		39	33	45	95	86	298
Uncrowded	13	6	3	5	7	3	24
	14	9	6	6	14	10	45
	15	2	2	6	15	8	33
	16	0	2	10	16	21	49
	17	3	2	0	3	14	22
	18	6	2	4	7	20	39
		26	17	31	62	76	212
Total		106	105	145	258	276	890

og de ligger i det såkaldte *lange format* på hjemmesiden.

Vi ønsker at undersøge, om boligforholdene har betydning for antallet af infektioner, samt om visse familiemedlemmer er mere udsat end andre.

1. *Indlæs data i det midlertidige datasæt* `swabs`*, og overvej strukturen i disse:*

   Vi indlæser i det lidt kortere navn `sw` og laver den numeriske `family` om til en faktor:

```
sw <-
read.table("http://publicifsv.sund.ku.dk/~lts/basal/data/swabs.txt",
header=T)

sw$family<-as.factor(sw$family)
```

   Herefter skaffer vi os et overblik ved at lave `summary`:

```
summary(sw)
```

```
> summary(sw)
     crowding      family       name        swabs
 crow    :30   1      : 5   child1:18   Min.   : 0.000
 overcrow:30   2      : 5   child2:18   1st Qu.: 5.000
 uncrow  :30   3      : 5   child3:18   Median : 8.500
               4      : 5   father:18   Mean   : 9.889
               5      : 5   mother:18   3rd Qu.:14.750
               6      : 5               Max.   :35.000
               (Other):60
```

   Vi har totalt set 90 observationer (18 familier, hver med 5 medlemmer), og 4 variable: `crowding, family, name` og `swabs`.

   (a) *Hvad er outcome?*

   Det er variablen `swabs`, som angiver antallet af infektioner over en vis periode. Vi skal analysere denne i normalfordelingsbaserede modeller, selv om dette måske ikke er helt rimeligt. Kommentarer til dette gives til sidst.

(b) *Hvilke kovariater har vi?*

Vi er interesserede i at evaluere effekten af boligforhold (`crowding`) samt status i familien (`name`). Men vi er også nødt til at tage hensyn til, at personerne hænger sammen 5 og 5 i familier.

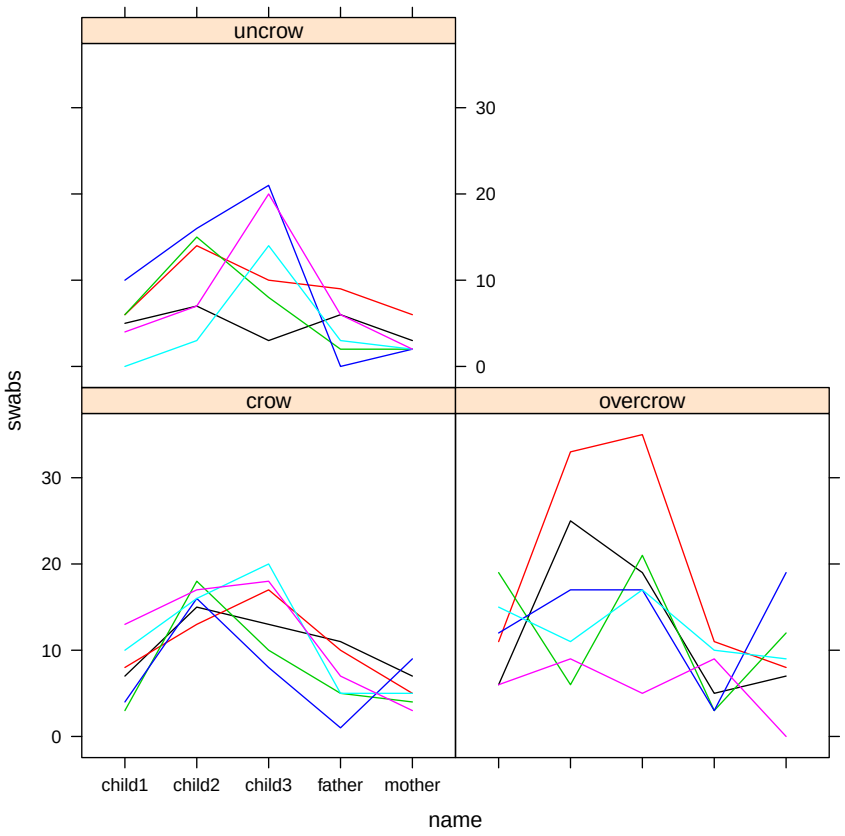
(c) *- og hvordan skal effekten af disse beskrives i modellen? dvs. hvilke er systematiske, og hvilke er tilfældige?*

Kovariaterne `crowding` og `name` er systematiske effekter, medens `family` er en tilfældig (random) effect, som er nestet i `crowding`. Vi er ikke interesserede i disse specifikke familier, de er bare repræsentanter for familier i al almindelighed.

2. *Lav en (eller flere) arbejdstegning(er), der så vidt muligt indeholder al informationen i data.*

Her er lavet tre spaghetti-plots, et for hver af de 3 typer af boligforhold (`crowding`), således at man kan se hver af de 18 familier. Denne konstruktion kræver installation af pakken `lattice`, og for at få tegnet linierne på fornuftig vis, sørger vi for at sortere dataframen først:

```
library(lattice)

sw <- sw[order(sw$crowding, sw$name),]

xyplot(swabs~name|crowding,group=family,type='l',data=sw)
```



Vi ser her generelt højere værdier for `overcrow`, måske mest for de mindste børn.

3. *Fit en passende varianskomponentmodel til disse data, dvs. en model, der specificerer samme korrelation mellem alle familiemedlemmer i samme familie.*

   Her er tale om en mixed model, med de ovenfor specificerede effekter, og vi vil benytte funktionen `lme`. Derfor skal vi først have fat i den ekstra pakke `nlme`:

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

For at få `uncrow` som reference, skriver vi

```
sw <- within(sw, crowding <- relevel(crowding, ref = "uncrow"))
```

hvorefter vi fitter den *additive* varianskomponentmodel:

```
model1 = lme(swabs ~ crowding + name, data=sw, random=~1 | family)
```

og får outputtet (beskåret)

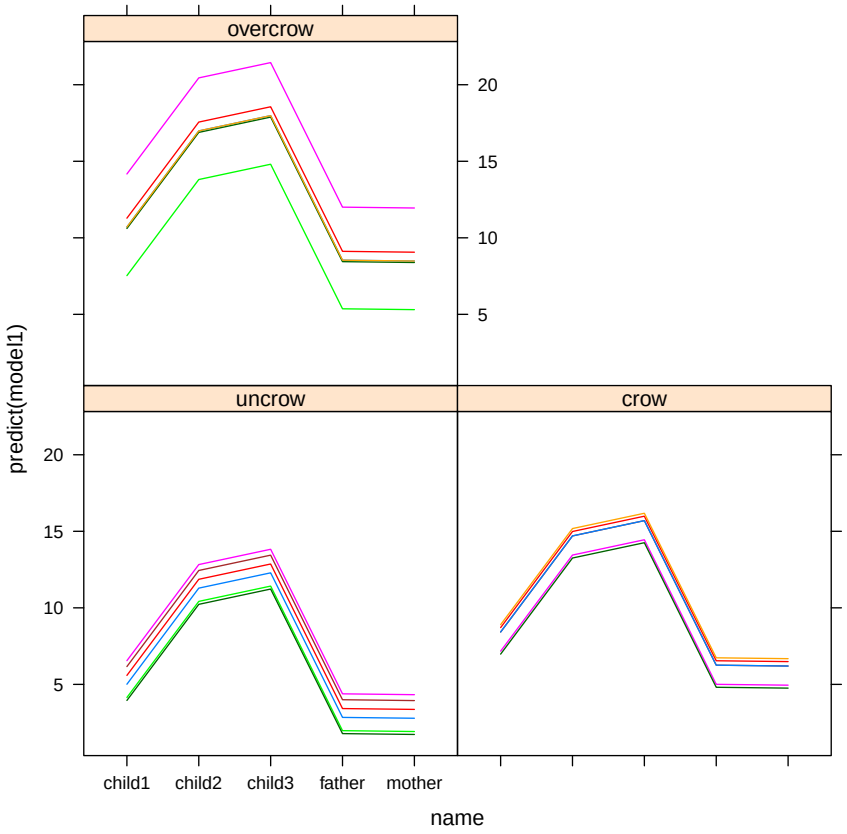
```
> summary(model1)
Linear mixed-effects model fit by REML
 Data: sw
       AIC      BIC    logLik
  545.1104 566.8799 -263.5552

Random effects:
 Formula: ~1 | family
        (Intercept) Residual
StdDev:    2.081828 4.834216

Fixed effects: swabs ~ crowding + name
                    Value Std.Error DF   t-value p-value
(Intercept)       5.233333  1.593730 68  3.283702  0.0016
crowdingcrow      2.866667  1.732814 15  1.654341  0.1188
crowdingovercrow  5.600000  1.732814 15  3.231737  0.0056
namechild2        6.277778  1.611405 68  3.895840  0.0002
namechild3        7.277778  1.611405 68  4.516417  0.0000
namefather       -2.166667  1.611405 68 -1.344582  0.1832
namemother       -2.222222  1.611405 68 -1.379059  0.1724

Standardized Within-Group Residuals:
        Min          Q1         Med          Q3         Max
-2.25198520 -0.69457063 -0.01723823  0.57199493  2.80356510

Number of Observations: 90
Number of Groups: 18
```

```
> intervals(model1)
Approximate 95% confidence intervals

 Fixed effects:
                      lower      est.     upper
(Intercept)       2.0530956  5.233333  8.413571
crowdingcrow     -0.8267398  2.866667  6.560073
crowdingovercrow  1.9065935  5.600000  9.293407
namechild2        3.0622685  6.277778  9.493287
namechild3        4.0622685  7.277778 10.493287
namefather       -5.3821759 -2.166667  1.048843
namemother       -5.4377315 -2.222222  0.993287
attr(,"label")
[1] "Fixed effects:"

 Random Effects:
  Level: family
                     lower     est.    upper
sd((Intercept)) 0.9684073 2.081828 4.475398

 Within-group standard error:
   lower     est.    upper
4.086371 4.834216 5.718924
```

```
> anova(model1)
            numDF denDF   F-value p-value
(Intercept)     1    68 195.40791  <.0001
crowding        2    15   5.22305   0.019
name            4    68  16.40661  <.0001
```

Den fittede model kan illustreres ved at tegne de predikterede værdier, som her er personspecifikke, dvs. de inkluderer de tilfældige effekter for hver familie:

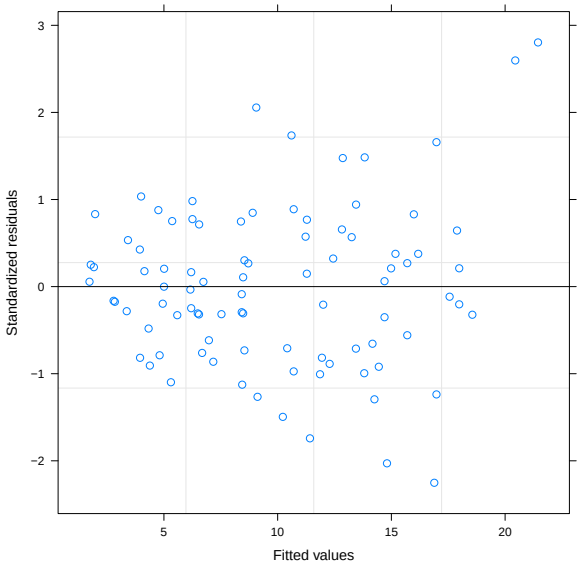
```
xyplot(predict(model1)~name|crowding,group=family,type='l',data=sw)
```



Fra outputtet ovenfor ses en signifikant effekt af `crowding` (P=0.019), og fra de tilhørende estimater ses, at der er flest infektioner, når man bor trangt og færre, når man bor bedre, som forventet.

Der er ligeledes en (stærkt) signifikant forskel på antal infektioner blandt de 5 typer af familiemedlemmer ($P < 0.0001$), idet de mindste børn har flest, og forældrene har færrest.

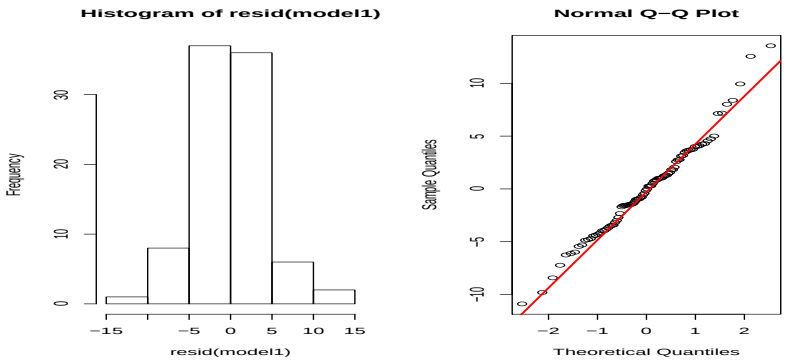
Den automatiske modelkontrol `plot(model1)` giver en tegning af residualer mod forventede værdier:



Her er tale om de *betingede* (conditional) residualer, hvor de person-specifikke predikterede værdier (se den tidligere figur) er fratrukket observationerne.

Vi kan desuden få et check af normalfordelingstilpasningen ved at skrive:

```
par(mfrow=c(1,2))
hist(resid(model1))
qqnorm(resid(model1))
qqline(resid(model1),col="red",lwd=2)
```

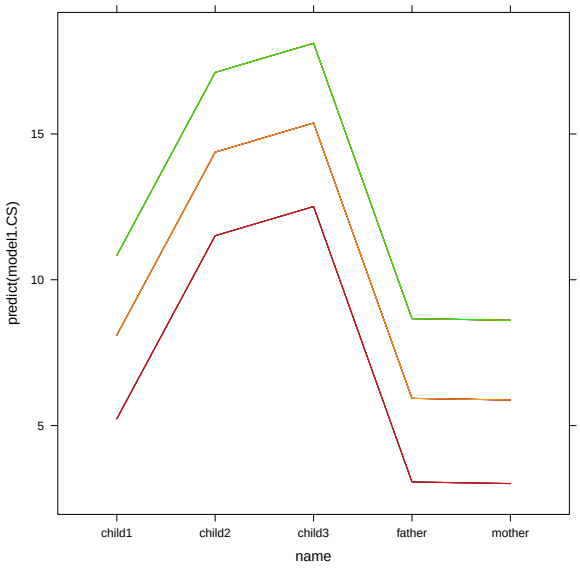


Hvis vi vil se på prediktioner på gruppeniveau, med tilhørende residualer, må vi i stedet fitte den samme model på en anden måde, nemlig med en Compound Symmetry kovariansstruktur:

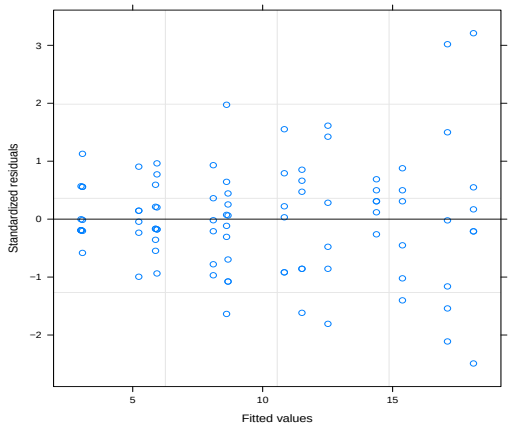
```
model1.CS = gls(swabs ~crowding + name, data=sw, correlation =
corCompSymm(form=~1 | family))
```

og derefter tegne de tilhørende fittede værdier med

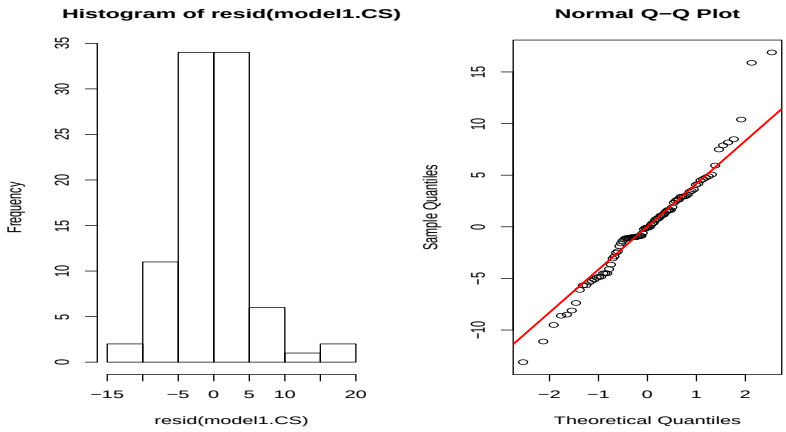
```
xyplot(predict(model1.CS)~name,group=family,type='l',data=sw)
```



Den automatiske modelkontrol `plot(model1.CS)` giver her en tegning af de *sædvanlige* residualer mod de *sædvanlige* forventede værdier:



samt checket for normalfordelingen:



På begge plots af residualer mod fittede værdier kan man se en tendens til trompetfacon, så modellen er ikke helt god. Mere om det til allersidst i besvarelsen.

(a) *Hvad er estimatet for denne korrelation mellem familiemedlemmer i samme familie?*

Korrelationen estimeres ud fra de to varianskomponenter, til

$$\frac{2.081828^2}{2.081828^2 + 4.834216^2} = 0.1564$$

(b) *Giv et estimat (med konfidensinterval) for forskellen på* `overcrowded` *og* `uncrowded`.

Da `uncrowded` er referenceniveau for `crowding`, kan vi direkte aflæse estimatet for denne sammenligning på linien `overcrow`. Det er 5.60(1.73), dvs. med 95% konfidensinterval på (1.91, 9.29).

(c) *Giv også et estimat (med konfidensinterval) for forskellen på ældste og yngste barn.*

Da det for `name` er det ældste barn, der er reference, får vi ligeledes direkte, at det yngste barn er mere belastet af infektioner end det ældste, med en forskel estimeret til 7.28 (1.61), med 95% konfidensinterval på (4.06, 10.49).

4. *Er der evidens for forskellig effekt af trange boligforhold for de enkelte familienmedlemmer?*

Ovenfor fittede vi en model, der havde en effekt af boligforhold, der var antaget at være den samme for alle familiemedlemmer, idet der var tale om en additiv model, dvs. uden interaktion (vekselvirkning).

Vi ser nu på en model *med* interaktion, for at se, om denne har signifikant betydning

```
model.int <- lme(swabs ~crowding + name + crowding*name,
             data=sw, random=~1 | family)
```

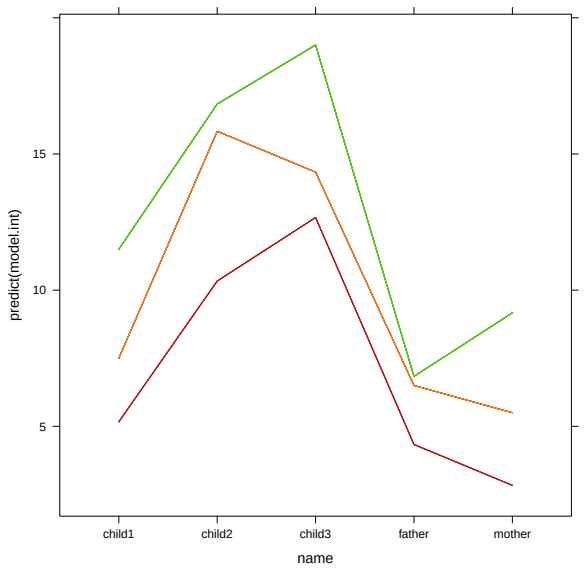
og vi finder P-værdierne:

```
> anova(model.int)

             numDF denDF   F-value p-value
(Intercept)      1    60 195.40651  <.0001
crowding         2    15   5.22301  0.0190
```

```
name              4    60  15.16747  <.0001
crowding:name     8    60   0.35801  0.9384
```

Der ses tydeligvis ingensomhelst indikation af interaktion, dvs. effekten af boligforhold synes at betyde nogenlunde det samme for alle typer af familiemedlemmer (P=0.94).

Figuren nedenfor illustrerer de predikterede værdier i modellen med interaktion (faktisk blot gennemsnit af 6 familier). Der ses en svag tendens til, at effekten af boligforhold er mindst for fædrene, men dette er som nævnt ovenfor, absolut ikke signifikant.



Figuren er dannet med

```
> model.int = gls(swabs ~crowding + name + crowding*name,
+ data=sw, correlation = corCompSymm(form=~1 | family))

xyplot(predict(model.int)~name,group=family,type='l',data=sw)
```

5. *Lav en illustration af de fittede værdier i den model, I ender med.*

Da der ikke er nogen signifikant interaktion, vil vi udelade interaktionsleddet igen, og ender således med modellen fra spørgsmål 3. De predikterede værdier i denne model har vi allerede set. Da dette er en additiv model, er der tale om 3 parallelle "kurver".

6. *Overvej, om det var fornuftigt, at benytte den anvendte kovariansstruktur. Prøv at fitte modellen med en ustruktureret kovarians og se, hvordan resultaterne ændrer sig.*

   Man kunne sagtens forestille sig, at der var større variation mellem nogle typer af familiemedlemmer (på tværs af familier) end mellem andre. Det oprindelige spaghettiplot i spørgsmål 2 tyder på en noget større variation mellem de små børn end f.eks. mellem fædre.

   Desuden kunne man forestille sig, at der var større korrelation mellem børnene indbyrdes, samt mellem mødre og børn, simpelthen fordi det muligvis kunne være begrænset, hvor meget faderen opholdt sig i familien.

   Vi kan undersøge disse ting ved at benytte en helt generel kovarinsstruktur i stedet for den "*compound symmetry*", som benyttes, når man ser det som en varianskomponentmodel.

   Vi udelader den ubetydelige interaktionen og opskriver den additive model med ustruktureret kovarians:

```
model.un = gls(swabs ~crowding + name, data=sw,
      corr = corSymm(form=~1 | family),
      weights = varIdent(form = ~ 1 | name), method="REML")
```

   og får herefter resultaterne

```
> summary(model.un)
Generalized least squares fit by REML
  Model: swabs ~ crowding + name
```

```
  Data: sw
       AIC      BIC    logLik
  546.8536 600.0681 -251.4268

Correlation Structure: General
 Formula: ~1 | family
 Parameter estimate(s):
 Correlation:
  1      2      3      4
2 -0.072
3  0.412  0.459
4 -0.079  0.053  0.041
5  0.344  0.053  0.144 -0.293
Variance function:
 Structure: Different standard deviations per stratum
 Formula: ~1 | name
 Parameter estimates:
   child1    child2    child3    father    mother
1.0000000 1.6355167 1.7684622 0.8757711 0.9561712

Coefficients:
                    Value Std.Error   t-value p-value
(Intercept)       5.573871  1.150124  4.846321  0.0000
crowdingcrow      2.723162  1.136230  2.396664  0.0188
crowdingovercrow  4.721892  1.136230  4.155755  0.0001
namechild2        6.277778  1.867879  3.360912  0.0012
namechild3        7.277778  1.544141  4.713156  0.0000
namefather       -2.166667  1.304210 -1.661286  0.1004
namemother       -2.222222  1.058903 -2.098608  0.0389

Standardized residuals:
        Min          Q1         Med          Q3         Max
-2.10669253 -0.64871569 -0.04664981  0.51274395  2.85112682

Residual standard error: 4.007996
Degrees of freedom: 90 total; 83 residual


> confint(model.un)
                     2.5 %     97.5 %
(Intercept)       3.3196691  7.8280728
crowdingcrow      0.4961921  4.9501309
crowdingovercrow  2.4949229  6.9488618
```

```
namechild2        2.6168020  9.9387535
namechild3        4.2513172 10.3042384
namefather       -4.7228717  0.3895384
namemother       -4.2976340 -0.1468105
```

```
> anova(model.un)
Denom. DF: 83
            numDF   F-value p-value
(Intercept)     1 211.47166  <.0001
crowding        2   8.70290   4e-04
name            4  10.52851  <.0001
```

I lighed med resultaterne fra spørgsmål 3, finder vi her, at der er en signifikant effekt af `crowding` (P=0.0004 mod før 0.019), og fra de tilhørende estimater ses, at der er flest infektioner, når man bor trangt og færre, når man bor bedre, som forventet.

Der er ligeledes en (stærkt) signifikant forskel på de 5 typer af familiemedlemmer ($P < 0.0001$), idet de mindste børn har flest, og forældrene har færrest.

I ouputtet ovenfor (under `Correlation:`) kan man se den estimerede korrelationsmatrix, men det er vanskeligt at overskue, i hvilken rækkefølge, de er skrevet ud. Vi benytter derfor følgende kommandoer for at udtrække informationen:

```
#Kovariansmatrix
Sigma <- getVarCov(model.un)
rownames(Sigma) <- unique(sw$name)
colnames(Sigma) <- unique(sw$name)

#Korrelationsmatrix
Corr <- cov2cor(getVarCov(model.un))
rownames(Corr) <- unique(sw$name)
colnames(Corr) <- unique(sw$name)
```

hvorefter vi udskriver de to matricer:

```
> Sigma
Marginal variance covariance matrix
        child1  child2  child3  father  mother
child1 16.0640 -1.8838 11.6920 -1.1163  5.2839
child2 -1.8838 42.9700 21.3110  1.2204  1.3430
child3 11.6920 21.3110 50.2400  1.0319  3.9190
father -1.1163  1.2204  1.0319 12.3210 -3.9356
mother  5.2839  1.3430  3.9190 -3.9356 14.6870
  Standard Deviations: 4.008 6.5551 7.088 3.5101 3.8323
```

```
> Corr
Marginal variance covariance matrix
          child1    child2   child3    father    mother
child1  1.000000 -0.071700 0.411580 -0.079349  0.344010
child2 -0.071700  1.000000 0.458660  0.053038  0.053459
child3  0.411580  0.458660 1.000000  0.041478  0.144280
father -0.079349  0.053038 0.041478  1.000000 -0.292570
mother  0.344010  0.053459 0.144280 -0.292570  1.000000
  Standard Deviations: 1 1 1 1 1
```

Fra diagonalen i variansmatricen (`Sigma`) ser vi, at der er mindst variation mellem fædre og størst variation mellem de yngste børn. Da det samme er tilfældet for selve niveauet af målingerne, kunne man fristes til at analysere logaritmer, men da der er ægte nuller i materialet, kan dette ikke lade sig gøre.

Fra korrelationsmatricen (`Corr`) ser vi, at de største korrelationer forekommer mellem mor og ældste barn (0.34), samt mellem yngste barn og dets søskende (0.41 resp. 0.46). Korrelationen mellem far og mor estimeres til at være negativ (-0.29), men nu skal man jo ikke overfortolke....

Vi kan sammenligne de to kovariansstrukturer ved at skrive

```
> anova(model.un, model1.CS)
          Model df      AIC      BIC    logLik   Test  L.Ratio p-value
model.un      1 22 546.8536 600.0681 -251.4268
model1.CS     2  9 545.1104 566.8799 -263.5552 1 vs 2 24.25681  0.0288
```

hvorved vi ser, at P-værdien bliver 0.29. Modellen med den mere generelle kovariansstruktur fitter altså signifikant bedre end den simple.

Estimatet for effekten af `overcrow` vs. `uncrow` ses at være 4.72 (1.14), hvor vi i spørgsmål 3 fik 5.60 (1.73). Grunden til, at vi får et lidt andet estimat end i den tidligere model er, at familiemedlemmerne nu vægter lidt anderledes i vurderingen af den enkelte families niveau, pga de uens varianser.

Tilsvarende er estimatet for forskellen på yngste og ældste barn 7.28 (1.54) mod før 7.28 (1.61). Her er det kun konfidensintervallet, der har ændret sig, fordi alle personer med samme status i familien vægtes ens i begge modeller.

7. *Hvilke betragtninger tror I, der ligger til grund for valget af design i denne undersøgelse? Et alternativ kunne jo være blot at undersøge et tilfældigt udpluk af personer.*

   Dette design giver bedre mulighed for at sammenligne familiemedlemmer, fordi der er tale om parrede sammenligninger. Herved slipper vi ved denne sammenligning for den store variation, man må forvente at finde mellem familier, pga. genetik, madvaner mv.

   Til gengæld får man en mere upræcis vurdering af effekten af boligforhold, fordi korrelationen mellem familiemedlemmer af samme familie nedsætter mængden af uafhængig information.

Hvis I herefter har tid, skal I prøve at se, om I kan opnå nogle af ovenstående resultater med traditionelle analysemetoder:

8. *Udregn gennemsnitligt antal infektioner for hver familie*

   Dette kunne sikkert gøres smartere, men her er valgt blot at gemme gennemsnittene i *den løse* variabel `mswabs`, som defineres ved

```
mswabs <- aggregate(sw$swabs, by=list(sw$family), FUN=mean)[,2]
```

   Herefter danner vi blot den tilhørende vektor `crowding` direkte, omdanner den til faktoren `Fcrowding` og definerer med det samme niveauet `uncrow` som reference til brug for den senere analyse:

```
crowding <- c(rep("overcrow",6), rep("crow",6), rep("uncrow",6))
Fcrowding <- factor(c(rep("overcrow",6), rep("crow",6), rep("uncrow",6)))
Fcrowding <- relevel(Fcrowding, ref = "uncrow")
```

Vi har nu 18 observationer, en for hver familie:

```
> cbind(crowding,mswabs)
      crowding   mswabs
 [1,] "overcrow" "12.4"
 [2,] "overcrow" "19.6"
 [3,] "overcrow" "12.2"
 [4,] "overcrow" "13.6"
 [5,] "overcrow" "12.4"
 [6,] "overcrow" "5.8"
 [7,] "crow"     "10.6"
 [8,] "crow"     "10.6"
 [9,] "crow"     "8"
[10,] "crow"     "7.6"
[11,] "crow"     "11.2"
[12,] "crow"     "11.6"
[13,] "uncrow"   "4.8"
[14,] "uncrow"   "9"
[15,] "uncrow"   "6.6"
[16,] "uncrow"   "9.8"
[17,] "uncrow"   "4.4"
[18,] "uncrow"   "7.8"
```

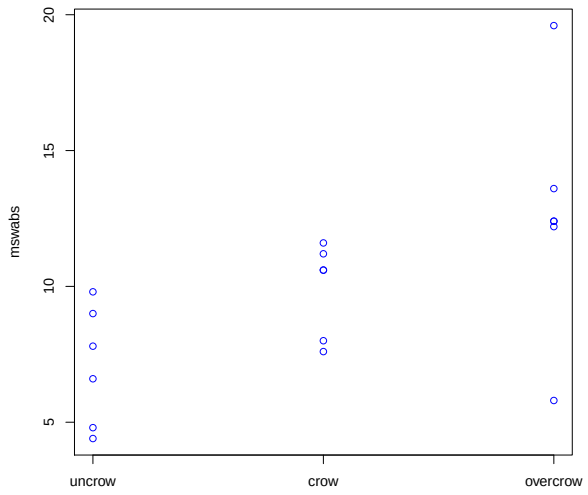
9. *Brug de nydannede data til at vurdere effekten af boligforhold*

   (a) *Hvilken type analyse er der tale om her?*
   Husk en figur til illustration.

   Vi skal sammenligne det gennemsnitlige antal infektioner i de 18 familier (`mswabs`), og familierne er inddelt i 3 grupper, angivet ved boligforholdene, `crowding`. Der er således tale om en ensidet variansanalyse.

   En passende figur kunne være et scatterplot, da vi har så få observationer (hvis vi havde haft flere, ville et Box Plot være mere rimeligt):

```
tekst <- c("uncrow","crow","overcrow")

par(mfrow=c(1,1))
plot(as.numeric(Fcrowding),mswabs, xaxt="n",xlab='',col="blue")
axis(1, at = 1:3, labels=tekst)
```



Nu laver vi den ensidede variansanalyse

```
model.mswabs <- lm(mswabs ~ Fcrowding)
```

som giver os outputtet

```
> summary(model.mswabs)

Call:
lm(formula = mswabs ~ Fcrowding)

Residuals:
   Min     1Q Median     3Q    Max
-6.867 -1.567  0.200  1.183  6.933

Coefficients:
                 Estimate Std. Error t value Pr(>|t|)
```

```
(Intercept)          7.067      1.225   5.767 3.72e-05 ***
Fcrowdingcrow        2.867      1.733   1.654  0.11883
Fcrowdingovercrow    5.600      1.733   3.232  0.00559 **
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 3.001 on 15 degrees of freedom
Multiple R-squared:  0.4105,Adjusted R-squared:  0.3319
F-statistic: 5.223 on 2 and 15 DF,  p-value: 0.01899
```

```
> confint(model.mswabs)
                        2.5 %   97.5 %
(Intercept)         4.4550248 9.678309
Fcrowdingcrow      -0.8267527 6.560086
Fcrowdingovercrow   1.9065807 9.293419
```

```
> anova(model.mswabs)
Analysis of Variance Table

Response: mswabs
           Df  Sum Sq Mean Sq F value  Pr(>F)
Fcrowding   2  94.098  47.049   5.223 0.01899 *
Residuals  15 135.120   9.008
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

Vi finder, ligesom i spørgsmål 3, at der er en signifikant effekt af boligforhold på antallet af infektioner, P=0.019.

(b) *Giv igen et estimat (med konfidensinterval) for forskellen på* `overcrowded` *og* `uncrowded`*, og sammenlign med det ovenfor fundne (spm. 3b).*

Da `uncrowded` er referenceniveau for `crowding`, kan vi igen direkte aflæse estimatet for denne sammenligning på linien `Fcrowdingovercrow`. Det er 5.60(1.73), dvs. med 95% konfidensinterval på (1.91, 9.29), altså ganske som vi fandt i "mixed effects"-modellen fra spørgsmål 3.

(c) *Hvorfor kan vi ikke udfra denne analyse sige noget om forskelle på familiemedlemmer?*

Vi regner her på gennemsnit over 5 familiemedlemmer, så vi kan derfor ikke sige noget om de indbyrdes forskelle på disse ud fra denne analyse.

10. *Nu ser vi et øjeblik væk fra faktoren* crowding *og betragter bare de 18 familier, som om de var tilfældige stikprøver fra samme population. Vi går altså tilbage til det oprindelige datasæt,* `sw`*:*

(a) *Lav en tosidet variansanalyse med faktorerne* `family` *og* `name`*.*

Her behøver vi blot at bruge `lm`:

```
anova2 <- lm(swabs ~ factor(family) + name, data=sw)
```

og finder outputtet:

```
> summary(anova2)

Call:
lm(formula = swabs ~ factor(family) + name, data = sw)

Residuals:
     Min       1Q   Median       3Q      Max
-10.6444  -2.5917   0.0056   2.5139   9.9556

Coefficients:
                   Estimate Std. Error t value Pr(>|t|)
(Intercept)       1.057e+01  2.390e+00   4.421 3.63e-05 ***
factor(family)2   7.200e+00  3.057e+00   2.355 0.021420 *
factor(family)3  -2.000e-01  3.057e+00  -0.065 0.948036
factor(family)4   1.200e+00  3.057e+00   0.392 0.695926
factor(family)5  -1.028e-16  3.057e+00   0.000 1.000000
factor(family)6  -6.600e+00  3.057e+00  -2.159 0.034408 *
factor(family)7  -1.800e+00  3.057e+00  -0.589 0.557992
factor(family)8  -1.800e+00  3.057e+00  -0.589 0.557992
factor(family)9  -4.400e+00  3.057e+00  -1.439 0.154704
factor(family)10 -4.800e+00  3.057e+00  -1.570 0.121069
factor(family)11 -1.200e+00  3.057e+00  -0.392 0.695926
factor(family)12 -8.000e-01  3.057e+00  -0.262 0.794376
factor(family)13 -7.600e+00  3.057e+00  -2.486 0.015391 *
factor(family)14 -3.400e+00  3.057e+00  -1.112 0.270034
factor(family)15 -5.800e+00  3.057e+00  -1.897 0.062073 .
factor(family)16 -2.600e+00  3.057e+00  -0.850 0.398093
factor(family)17 -8.000e+00  3.057e+00  -2.617 0.010934 *
```

```
factor(family)18 -4.600e+00  3.057e+00  -1.505 0.137075
namechild2        6.278e+00  1.611e+00   3.896 0.000226 ***
namechild3        7.278e+00  1.611e+00   4.516 2.57e-05 ***
namefather       -2.167e+00  1.611e+00  -1.345 0.183228
namemother       -2.222e+00  1.611e+00  -1.379 0.172396
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 4.834 on 68 degrees of freedom
Multiple R-squared:  0.6277,Adjusted R-squared:  0.5128
F-statistic:  5.46 on 21 and 68 DF,  p-value: 4.356e-08
```

```
> anova(anova2)
Analysis of Variance Table

Response: swabs
              Df Sum Sq Mean Sq F value    Pr(>F)
factor(family) 17 1146.1   67.42  2.8848   0.00102 **
name            4 1533.7  383.42 16.4066 1.874e-09 ***
Residuals      68 1589.1   23.37
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

Her ser vi en tydelig signifikant forskel på de 18 familier (P=0.001), men en endnu tydeligere forskel på familiemedlemmerne ($P < 0.0001$).

(b) *Er der evidens for forskelle på de 5 familiemedlemmer?*

Ja, som nævnt ovenfor er dette meget tydeligt ($P < 0.0001$). De yngste børn ser ud til at være de mest sensitive.

(c) *Giv igen et estimat (med konfidensinterval) for forskellen på ældste og yngste barn, og sammenlign med det tidligere fundne (spm. 3c).*

Her finder vi ligesom før, at det ældste barn er mindre belastet af infektioner end det yngste, med en forskel estimeret til 7.28 (1.61), ganske som i spørgsmål 3c.

11. *Kan vi vurdere interaktionen mellem* `crowding` *og* `name` *uden at benytte varianskomponentmodellen?*

Ja, det kan man faktisk godt, men det kræver, at man bibeholder familien som *fixed effect*, altså benytter koden

```
anova3 <- lm(swabs ~ factor(crowding) + factor(family) + name
+factor(crowding)*name, data=sw)
```

der vil give outputtet

```
> anova(anova3)
Analysis of Variance Table

Response: swabs
                      Df  Sum Sq Mean Sq F value    Pr(>F)
factor(crowding)       2  470.49  235.24  9.3060 0.0003019 ***
factor(family)        15  675.60   45.04  1.7817 0.0589999 .
name                   4 1533.67  383.42 15.1675 1.267e-08 ***
factor(crowding):name  8   72.40    9.05  0.3580 0.9383526
Residuals             60 1516.73   25.28
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

som ses at give det samme resultat som i spørgsmål 4.

12. *Diskuter hvad der sker med de forskellige analysetyper, hvis der mangler nogle observationer.*

    Det afhænger meget af, *hvorfor* sådanne observationer mangler.

    Hvis vi bare er kommet til at smide informationen væk, vil "mixed model" stadig give fornuftige svar og udnytte alle forhåndenværende observationer. Men den ensidede variansanalyse på gennemsnittene for hver familie vil ikke længere give det samme resultat, den vil faktisk være *forkert*. Hvor meget forkert, den er, afhænger af, hvilken observation, der mangler. Hvis f.eks. det yngste barn fra en overcrowded familie mangler, vil denne familie se ud til at have et ret lavt antal gennemsnitlige infektioner, og derved vil effekten af boligforhold blive undervurderet.

Hvis en observation mangler *fordi* den f.eks. er meget høj, så er der ikke nogen måde at redde det på! Det kaldes *informative missing*, og det er anledning til mange fejl og mistolkninger.

Konklusioner:

- Der er en signifikant forskel på familier (P=0.01), men det interessrerer os sådan set ikke. Denne faktor optræder i modellen som en tilfældig effekt, som gør det muligt at vurdere effekten af boligforhold.
- Der er en effekt af boligforhold (`crowding`) på antallet af infektioner (P=0.019), og effekten af denne kan kvantificeres med `proc mixed`, eller ved hjælp af ensidet variansanalyse på simple familie-gennemsnit, det sidste dog **kun** i tilfælde af fuldstændigt datasæt, dvs. **ingen manglende værdier**.
- Der er signifikant forskel på antallet af infektioner hos de forskellige typer af familiemedlemmer ($P < 0.0001$). Der er flest infektioner blandt de yngste børn, og færrest hos forældrene.

Men:
Vi har hele vejen antaget, at det er rimeligt at lave normalfordelingsbaserede analyser, og dette ser ikke rigtigt fornuftigt ud, f.eks. fordi et antal altid er ikke-negativt, og her har vi endda en del 0'er.

Man kunne forsøge sig med en kvadratrodstransformation, men det giver vanskeligheder ved fortolkningen.

Hvis man ønsker at udregne prediktionsområder for antallet af infektioner, skal man i hvert fald gøre et eller andet anderledes end ovenfor, og dette kunne være at modellere antallene som Poissonfordelte. Dette er dog ikke en del af dette kursus.