Opgave 1.1

```
> ## (a)
>  10+5*3
[1] 25
>  8/2*2  ## farlig mangel paa parenteser!
[1] 8
>  8/(2*2) ## osv.
[1] 2
> ## (b)
>  0.5^6
[1] 0.015625
>  0.5**6 ## samme som linien foer.
[1] 0.015625
> ## (d) og (e)
>  bio=41.9
>  bio0= 211.5
>  lnQ= log(bio/bio0) ## log(x) er nat. log. til x
>  lnQ  ## for at udskrive resultatet
[1] -1.618939
>  print(lnQ) ## samme
[1] -1.618939
> ## (f) og (g)
>  dif.ln.bio = log(bio) - log(bio0)
>  dif.ln.bio - lnQ ## =0, så godt som maskinen kan ramme det!
[1] -4.440892e-16
>  dif.ln.bio== lnQ ## bliver FALSE, pga. afrundingsfejl!
[1] FALSE
> ## (h)
>  log10(bio)
[1] 1.622214
>  log(bio, base=10) ## bemaerk det ekstra argument til funktionen.
[1] 1.622214
```

Opgave 1.2

I web-browseren klikker du blot på datasættet for at se det på skærmen. Vi viser først hele R-kørslen. Nogle kommentarer følger dernæst.

```
> ## (b)
>  potter = read.table(file.choose(), header=TRUE)
> ## (d)
>  attach(potter)
>  Nitrogen
```

```
 [1] 19.4 32.6 27.0 32.1 33.0 17.7 24.8 27.9 25.2 24.3 17.0 19.4  9.1 11.9 15.8
[16] 20.7 21.0 20.5 18.8 18.6 14.3 14.4 11.8 11.6 14.2 17.3 19.3 19.1 16.9 20.8
>  Treat
 [1] A A A A A B B B B B C C C C C D D D D D E E E E E F F F F F
Levels: A B C D E F
> ## (e)
>  mean(Nitrogen) ## osv.
[1] 19.88333
> ## (f)
>  Nitrogen3= subset(Nitrogen, Nitrogen>25)
> Nitrogen3
[1] 32.6 27.0 32.1 33.0 27.9 25.2
> ## (g)
>  model = lm(Nitrogen ~ Treat) ## da Treat allerede er en factor - ellers factor(Treat)
>  anova(model)
Analysis of Variance Table

Response: Nitrogen
          Df Sum Sq Mean Sq F value    Pr(>F)
Treat      5 847.29  169.46  14.381 1.475e-06 ***
Residuals 24 282.80   11.78
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
>
>  summary(model)

Call:
lm(formula = Nitrogen ~ Treat)

Residuals:
   Min     1Q Median     3Q    Max
-9.420 -1.440  0.700  1.205  4.760

Coefficients:
            Estimate Std. Error t value Pr(>|t|)
(Intercept)   28.820      1.535  18.774 7.49e-16 ***
TreatB        -4.840      2.171  -2.229  0.03541 *
TreatC       -14.180      2.171  -6.532 9.36e-07 ***
TreatD        -8.900      2.171  -4.099  0.00041 ***
TreatE       -15.560      2.171  -7.167 2.09e-07 ***
TreatF       -10.140      2.171  -4.671 9.59e-05 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 3.433 on 24 degrees of freedom
Multiple R-Squared: 0.7498,     Adjusted R-squared: 0.6976
F-statistic: 14.38 on 5 and 24 DF,  p-value: 1.475e-06
> ## (h)
```

```
>  coef(model)
(Intercept)      TreatB      TreatC      TreatD      TreatE      TreatF
      28.82       -4.84      -14.18       -8.90      -15.56      -10.14
>  resid(model)
    1     2     3     4     5     6     7     8     9    10    11    12    13
-9.42  3.78 -1.82  3.28  4.18 -6.28  0.82  3.92  1.22  0.32  2.36  4.76 -5.54
   14    15    16    17    18    19    20    21    22    23    24    25    26
-2.74  1.16  0.78  1.08  0.58 -1.12 -1.32  1.04  1.14 -1.46 -1.66  0.94 -1.38
   27    28    29    30
 0.62  0.42 -1.78  2.12
>  confint(model)
                2.5 %     97.5 %
(Intercept)  25.651641  31.988359
TreatB       -9.320736  -0.359264
TreatC      -18.660736  -9.699264
TreatD      -13.380736  -4.419264
TreatE      -20.040736 -11.079264
TreatF      -14.620736  -5.659264
>
> ## (i)
>  plot(predict(model), resid(model))
>  abline(h=0) ## tilføjer 0-linien
>
> ## (j)
>  model2= lm(Nitrogen ~ 1)
>  summary(model2)

Call:
lm(formula = Nitrogen ~ 1)

Residuals:
     Min       1Q   Median       3Q      Max
-10.7833  -3.8083  -0.6833   3.5917  13.1167

Coefficients:
            Estimate Std. Error t value Pr(>|t|)
(Intercept)    19.88       1.14   17.45   <2e-16 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 6.242 on 29 degrees of freedom

>
> ## (k)
>  anova(model2, model)
Analysis of Variance Table

Model 1: Nitrogen ~ 1
```

```
Model 2: Nitrogen ~ Treat
  Res.Df     RSS Df Sum of Sq      F    Pr(>F)
1     29 1130.08
2     24  282.80  5    847.29 14.381 1.475e-06 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

I udskriften vedrørende spørgsmål (d) skal det bemærkes af `Levels: A, ..` viser at R allerede har kodet `Treat` som en faktor (fordi værdierne er "text").

I (g) ses testen for hypotesen om ingen behandlingseffekt at kunne afvises meget klart. Der er altså med stor sikkerhed effekt af behandlingerne. Med `summary(model)` ses blandt andet estimater for behandlingseffekterne og de tilhørende standard errors. Vi ser at behadnling A giver højest nitrogen, mens C og E giver lavest nitrogen. Bemærk at listen af estimater også fås som vektor i (h) med `coef(model)`. Her giver `resid(model)` residualerne og `confint(model` giver konfidensintervaller for de samme parametre som ses med `summary`. Hered får vi i dette tilfælde udskrevet konfidensintervaller for middelværdien af nitrogen med behandling A, samt for differencen mellme middlværdien af hver af de øvrige behandlinger og behandling A.

I (i) får vi det almindeligt benyttede residualplot, som ikke viser noget særlig betænkeligt mht. modelantagelser her. I (k) får vi samme test som i spørgsmål (g) med `anova(model`, men i mere komplicerede modeller, er det bedre at benytte formen fra (k) hvor man styrer hvilken model man tester mod en hvilken anden. (Den "lille" model skal stå først).

Opgave 1.3

```
## (a)

> udb = scan("fosfor-udbytter.txt", skip=1)
> beh = rep(1:9, each=3)

> udb
 [1] 330 320 355 346 350 369 409 363 414 368 340 366 371 373 403 409 410 402 360
[20] 396 406 382 407 425 398 415 433
> beh
 [1] 1 1 1 2 2 2 3 3 3 4 4 4 5 5 5 6 6 6 7 7 7 8 8 8 9 9 9

## (b)

> plot(beh, udb)

## (c) Not really relevant since there might be a treatment effect.

> boxplot(udb)

## (d) Box-plot for each treatment.

> behfac = factor(beh)
```

```
> plot(behfac, udb)

## (e) The plot from (d) is more appropriate since we expect different
## distribution for different treatments. Graphical comparison of the
## nine treatments.

## (f)

> m1 = lm(udb ~ behfac)

## (g) Summary uses treatment 1 as the reference level and compares the
## other treatments to this one. predict gives the expected yield for
## each observation.

> summary(m1)

Coefficients:
            Estimate Std. Error t value Pr(>|t|)
(Intercept)   335.00      10.90  30.730  < 2e-16 ***
behfac2        20.00      15.42   1.297 0.210916
behfac3        60.33      15.42   3.913 0.001018 **
behfac4        23.00      15.42   1.492 0.153051
behfac5        47.33      15.42   3.070 0.006595 **
behfac6        72.00      15.42   4.670 0.000190 ***
behfac7        52.33      15.42   3.395 0.003230 **
behfac8        69.67      15.42   4.519 0.000266 ***
behfac9        80.33      15.42   5.211  5.9e-05 ***

> predict(m1)
       1        2        3        4        5        6        7        8
335.0000 335.0000 335.0000 355.0000 355.0000 355.0000 395.3333 395.3333
       9       10       11       12       13       14       15       16
395.3333 358.0000 358.0000 358.0000 382.3333 382.3333 382.3333 407.0000
      17       18       19       20       21       22       23       24
407.0000 407.0000 387.3333 387.3333 387.3333 404.6667 404.6667 404.6667
      25       26       27
415.3333 415.3333 415.3333

## (h)

> anova(m1)
Analysis of Variance Table

Response: udb
          Df  Sum Sq Mean Sq F value    Pr(>F)
behfac     8 17909.3  2238.7  6.2792 0.0006054 ***
Residuals 18  6417.3   356.5
```

```
## (i)

> fosfordat = read.table("fosfor.txt", header=T)
> attach(fosfordat)

## (j)

> p82fac = factor(p82)
> model82 = lm(udbytte ~ p82fac)

## (k) Cannot compare m1 and model82 immediately since they different
## names of the response have been used.  We fit the full model again,
## now with the new variable names, and test model82 against it. We
## reject that the yield depends only on the 1982-treatment (p=0.01).

> fosforfac = factor(fosfor)
> model1 = lm(udbytte ~ fosforfac)

> anova(model82, model1)
Analysis of Variance Table

Model 1: udbytte ~ p82fac
Model 2: udbytte ~ fosforfac
  Res.Df     RSS Df Sum of Sq      F  Pr(>F)
1     24 14863.8
2     18  6417.3  6    8446.4 3.9486 0.01076 *
```

Opgave 1.4

a) Der er benyttet en model for ensidet variansanalyse, dvs.

$$Y_i = \mu + \alpha(N_i) + e_i$$

hvor Y_i er tørvægt for den i'te potte ($i = 1, \ldots, 16$), μ, $\alpha(0.5)$, $\alpha(1)$, $\alpha(2)$ og $\alpha(3)$ er parametre, og e_i'erne er uafhængige og normalfordelte med middelværdi 0 og spredning σ.

Bemærk måden modellen skrives i R: et led af formen `factor(N)` betyder at modellen indeholder et led som afhænger vilkårligt af niveauet af den pågældende faktor. Hvis `N` forvejen var en faktor i R, kunne man blot have skrevet modellen som `dw ~ N`. Bemærk også at konstantleddet μ automatisk kommer med i R (kaldet "intercept") med mindre man beder om at det udelades (se senere i opgaven).

b) 9.47 er F-teststørrelsens værdi som ved beregning i F-fordelingen giver P-værdien 0.0017 som med stor sikkerhed viser at N-niveauerne ikke giver samme middel-tørvægt (α'erne er ikke ens). Residual-MS på 13.23 er det samme som s^2, hvilket betyder at estimatet for residual-spredningen er $\hat{\sigma} = s = \sqrt{13.23} = 3.64$. Det tilhørende frihedsgradstal (df) er 12.

c) Den prædikterede værdi for den enkelte gruppe fås ved at lægge de bidrag sammen som hører til gruppen og som ses i udskriften under estimat. Da konstantleddet indgår i alle grupper (det er derfor det kaldes konstant!) får vi

N-niveau	Prædikteret værdi	Estimat for
0.5	24.465	$\mu + \alpha(0.5)$
1.0	24.465 + 7.635	$\mu + \alpha(1.0)$
2.0	24.465 + 10.118	$\mu + \alpha(2.0)$
3.0	24.465 + 13.073	$\mu + \alpha(3.0)$

idet R har udeladt første faktor-niveau hvilket betyder at dette faktorniveau kommer til at svare til interceptet (det er valgt som "referenceniveau").

d) Af tabellen ovenfor ses at 7.635 er estimatet for $\alpha(1.0) - \alpha(0.5)$ så det derved repræsenterer forskellen på middelværdien med N-niveau 1.0 og N-niveau 0.5 (forskellen til referencen).

e) Konfidensintervallet bliver

$$\alpha(1.0) - \alpha(0.5) : 7.635 \pm 2.179 \cdot 2.572 = 7.6 \pm 5.6$$

hvor t-fraktilen på 2.179 fås fra R, og hvor vi, som man *altid* skal, har benyttet den standard error som hører til estimatet. Vi får LSD = 5.6, idet LSD-værdien pr. definition er tallet efter $\pm$ i konfidensintervallet for forskellen mellem de prædikterede værdier for to faktorniveauer. Af formlen i kompendiet fremgår dels at standard error for forskellen kan beregnes som

$$\text{se}(\hat{\alpha}(1.0) - \hat{\alpha}(0.5)) = s\sqrt{\frac{1}{4} + \frac{1}{4}} = 2.572,$$

dels at LSD-værdien er den samme uanset hvilke to grupper der sammenlignes, idet der er 4 observationer i hver gruppe.

f) Her er modellen skrevet uden intercept, dvs. som

$$Y_i = \mu(N_i) + e_i$$

hvilket betyder at vi har samme model som før men med omskrivningen $\mu(j) = \mu + \alpha(j)$ for $j = 1, 2, 3, 4$. I udskriften er angivet estimaterne for de fire $\mu(j)$'er, altså gruppemiddelværdierne. Dette kan være mere praktisk end første udskrift, når man er interesseret i *estimaterne*, derimod er første modelopskrivning mere praktisk hvad angår test, idet den tester forskelle på grupperne, hvorimod der i anden udskrift kommer test for om de enkelte gruppemiddelværdier kan være nul, hvilket er uden praktisk relevans.

Opgave 1.5

a) Vi benytter modellen for lineær regressionsanalyse, altså en retlinet sammenhæng mellem X og Y suppleret med tilfældig normalfordelt variation. Brugen af denne model er baseret på et punktdiagram (X-Y plot) som ikke er vist her. Modellen er

$$Y_i = \alpha + \beta X_i + e_i$$

for ar $i = 1, \ldots, 14$, hvor α og β er parametre, og e_i'erne er uafhængige og normalfordelte med middelværdi 0 og spredning σ.

Bemærk at i modelopskrivningen i R er konstantleddet, α, automatisk med (også kaldet intercept), mens `X` som *ikke* er specificeret som en faktor derved indgår med sin talværdi ganget med en konstant β (ofte kaldet hældningen hørende til X).

b)

$$\hat{\alpha} = -1.58, \quad \hat{\beta} = 0.974, \quad \hat{\sigma} = s = 0.325.$$

Det er α og β der specificerer sammenhængen. Konfidensintervallerne er

$$\alpha: \quad -1.58 \pm 2.179 \cdot 0.219 = -1.58 \pm 0.48$$

og

$$\beta: \quad 0.974 \pm 2.179 \cdot 0.0377 = 0.97 \pm 0.082$$

frihedsgradstal (df) er 12.

c) Den systematisk del af modellen er

$$\ln F = \alpha + \beta \ln W$$

som ved at tage eksponentialfunktionen på begge sider kan omskrives til

$$F = e^{\alpha} \cdot W^{\beta}$$

Hvis $\beta = 1$ svarer dette netop til at F udgør en fast andel af kropsvægten, nemlig e^{α}. Vi tester derfor hypotesen $\beta = 1$ ved brug af en t-test som giver $t = -0.685$ (se udskriften) svarende til P-værdien

$$P = 2 \cdot 0.253 = 0.51.$$

Data passer altså pænt med at vægten af flyvemusklerne udgør en fast andel af kropsvægten.

d) Prædiktionsintervallet for $\ln F$ baseret på kropsvægten 2.7 gram og modellen for de 14 øvrige arter er $(-1.42, 0.21)$ og den prædikterede værdi af $\ln F$ er -0.61. Den faktiske værdi, $X = \ln 1.0 = 0.0$ ses at ligge inden for dette interval og kan dermed ikke siges at være specielt afvigende fra mønstret fra de øvrige arter. Vi ser dog at den faktisk vægt af flyvemusklerne ligger i den høje ende af intervallet hvilket stemmer godt overens med kolibriens gode flyveegenskaber.

e) Vi vil gerne se

- Et punktdiagram af Y mod X for at kontrollere rimeligheden af den systematiske del af modellen (den rette linie). Det produceres i R ved at skrive `plot(X,Y)`.
- Et plot af residualer mod prædikterede værdier for at vurdere antagelsen om varianshomogenitet og (igen) for at se efter systematiske afvigelser fra modellen (den rette linie).
- Et qq-plot over residualerne (eller eventuelt et histogram) for at vurdere antagelsen om normalfordeling.

I R skriver man blot

```
plot(res)
```

hvorefter der kommer fire tegninger efter tur, heriblandt de to sidst nævnte. Hvis man vil have de fire tegninger at se samtidigt som fire figurer sat op i to rækker og to søjler, skriver man

```
par(mfrow=c(2,2))
plot(res)
```

Opgave 1.8

Opgave 1.9

Vi viser det komplette R-program, men uden udskriften.

```
## (a)
 fosfor = read.table(file.choose(), header=TRUE)
 table(fosfor$fosfor)
## (b)
 table(fosfor$p81, fosfor$p82)
## (c)
 data.kun.81= subset(fosfor, p82==0)
 data.kun.81 ## voila!
## (d)
 mean(data.kun.81$udbytte)
 boxplot(data.kun.81$udbytte)
## (e)
 boxplot(fosfor$udbytte ~ factor(fosfor$p82))
## (f)
 ?head
 head(fosfor, n=10)
## (g)
 which(fosfor$p82==0) ## NB: dobbelt lighedstegn ved sammenligninger!
## (h)
 which(fosfor$p82==0 & fosfor$p81==0)
## (i)
 data.uden.8 = fosfor[-8,] ## matrixnotation: (rk uden 8, alle soejler)
 dim(data.uden.8) ## viser at der nu kun er 26 raekker
```