Regression Models in Systems Biology with R

Part II: General Linear Model

Uwe Menzel 2014

www.matstat.org

Outline

1. Simple Linear Regression
    1. The statistics behind the output of "**`lm`**"
2. **General Linear Model**
    1. Continuous and categorical variables mixed, "**`lm`**"
    2. Interaction
3. Generalized Linear Model
    1. Logistic Regression - "**`glm`**"
    2. Multinomial Regression - "**`multinom`**"

2. General Linear Model

A general linear model includes multiple independent variables.

$$y = \beta_0 + \beta_1 \cdot x_1 + \beta_2 \cdot x_2 + \ldots + \beta_k \cdot x_k + \varepsilon \qquad \varepsilon \sim N(0, \sigma)$$

We have k independent variables (and still one dependent variable). Because we have N measurements for each independent variable, and N measurements for the dependent variable, the x_k and y should now be written as vectors. For the i-th measurement, we can write:

$$y_i = \beta_0 + \beta_1 \cdot x_{i1} + \beta_2 \cdot x_{i2} + \ldots + \beta_k \cdot x_{ik} + \varepsilon_i \qquad \varepsilon_i \sim N(0, \sigma)$$

Regarding the x- variables, the first index stands for the measurement, the second index indicates the variable. This can also be written (here for 3 measurements and 3 independent variables):

$$\begin{pmatrix} y_1 \\ y_2 \\ y_3 \end{pmatrix} = \beta_0 + \beta_1 \begin{pmatrix} x_{11} \\ x_{21} \\ x_{31} \end{pmatrix} + \beta_2 \begin{pmatrix} x_{12} \\ x_{22} \\ x_{32} \end{pmatrix} + \beta_3 \begin{pmatrix} x_{13} \\ x_{23} \\ x_{33} \end{pmatrix} + \begin{pmatrix} \varepsilon_1 \\ \varepsilon_2 \\ \varepsilon_3 \end{pmatrix}$$

Simulation of Multidimensional data

```
n = 10 # sample size

x1 = runif(n, 0, 100)
x2 = runif(n, 10, 200)
x3 = runif(n, 100, 400)

cor.test(x1,x2) # p-value = 0.2619  OK, not sign. correlated
cor.test(x1,x3) # p-value = 0.3302  OK, not sign. correlated
cor.test(x2,x3) # p-value = -0.1205 OK, not sign. correlated
```

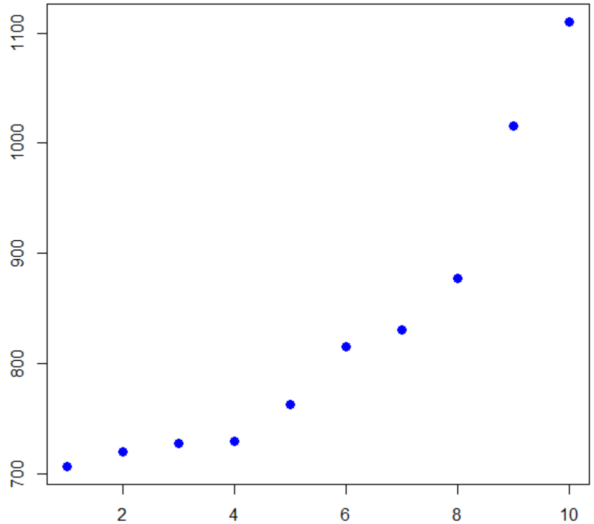
The x_i must not be (strongly) correlated! (use also `pairs` function in R)
If the predictors were correlated, the model wouldn't know how to "distribute" the regression coefficients between them (→ "`NA`" for estimated coefficients)

```
y = 3 + 2*x1 + 3*x2 + 1*x3 + rnorm(n, 0, 2) # simulated response
```

Let's see if we can "rediscover" the true coefficients chosen above by regression!

Multidimensional Regression with "`lm`"

```
mdata = data.frame(y = y, x1 = x1, x2 = x2, x3 = x3)
mdata = mdata[order(mdata$y),] # sort according to y
head(mdata)
#           y       x1        x2       x3
# 4  437.4030 10.60656  10.94048  378.8306
# 10 588.1563 31.24962  36.42743  412.4942
# 6  629.5175 73.75266  89.65545  208.8536
# 8  653.3158 85.38278  97.99875  185.5635
# 5  739.1781 56.60325 118.66963  269.5035
# 3  739.7772 49.55353 179.67645  100.9213
plot(mdata$y)
```



see General_Reg_Models_Examples.R

Multidimensional Regression with "lm"

$$y = \beta_0 + \beta_1 \cdot x_1 + \beta_2 \cdot x_2 + \ldots + \beta_k \cdot x_k + \varepsilon \qquad \varepsilon \sim N(0, \sigma)$$

```
lm.res = lm(y ~ x1 + x2 + x3, data = mdata) # Additive model
# Call:
# lm(formula = y ~ x1 + x2 + x3, data = mdata)
# Coefficients:
# (Intercept)     x1     x2     x3
#       3.153 2.027 2.974 1.003
```

- "Additive model"
- Wilkinson-Rogers Notation, translates to the above model
- The coefficients were successfully rediscovered.

More output using "`summary`"

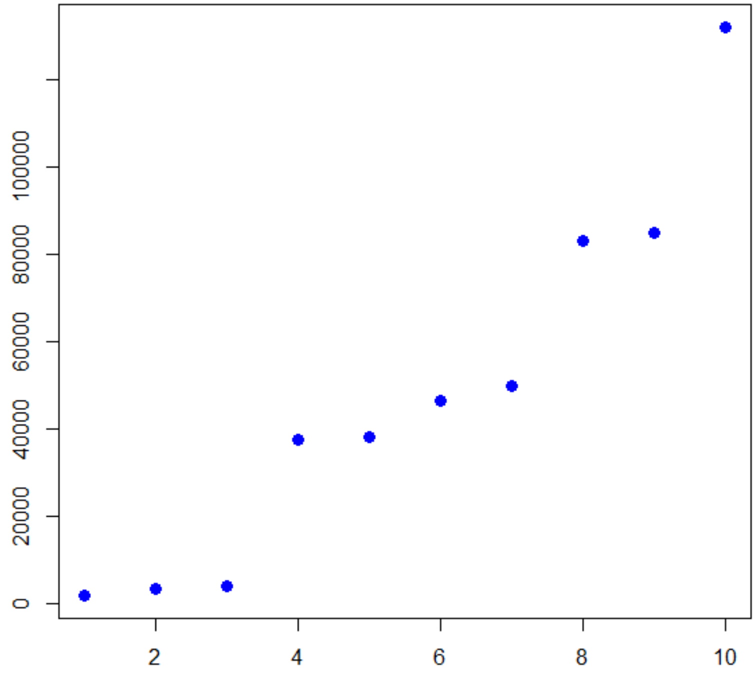
```
summary(lm.res)
# Call:
# lm(formula = y ~ x1 + x2 + x3, data = mdata)
# Residuals:
#      Min       1Q   Median       3Q      Max
# -1.76126 -0.94692 -0.04002 0.65184 2.76677
# Coefficients:
#              Estimate  Std. Error  t value   Pr(>|t|)
# (Intercept)  3.152911    2.658689    1.186       0.28
# x1           2.026939    0.025607   79.155   2.74e-10 ***
# x2           2.973589    0.010156  292.797   1.07e-13 ***
# x3           1.002512    0.005698  175.954   2.27e-12 ***
# Residual standard error: 1.632 on 6 degrees of freedom
# Multiple R-squared: 1, Adjusted R-squared: 0.9999
# F-statistic: 4.486e+04 on 3 and 6 DF, p-value: 1.938e-12
```

- Output analogous to simple linear regression (t-tests), but $F \neq t^2$
- H_0 for F-test: $\beta_1 = \beta_2 = \ \ldots = \ \beta_p = 0$
  - "is there some dependence between the x_i and y ?
- $R^2 = 1$ very good model for the data obtained (weak noise)
- Extractor functions: `coef(lm.res)`, `resid(lm.res)`, `anova(lm.res)`,…

Multiple Linear Regression with Interaction

Simulate new response variable:

```
y = 3 + 2*x1 + 3*x2 + 1*x3 + 4*x1*x3 + rnorm(n, 0, 2) # interaction!
mdata = data.frame(y = y, x1 = x1, x2 = x2, x3 = x3)
mdata = mdata[order(mdata$y),]
plot(mdata$y)
```



Despite of the non-linearity in x, the model is still linear w.r.t. the β_i → multiple **linear** regression can be applied !

Multiple Linear Regression with Interaction

Simulated data:

```
y = 3 + 2*x1 + 3*x2 + 1*x3 + 4*x1*x3 + rnorm(n, 0, 2) # interaction!
```

... corresponds to the model:

$$y = \beta_0 + \beta_1 \cdot x_1 + \beta_2 \cdot x_2 + \beta_3 \cdot x_3 + + \beta_4 \cdot x_1 \cdot x_3 + \varepsilon$$

... translates to Wilkinson-Rogers-Notation:

```
y ~ x1 + x2 + x3 + x1:x3   # interaction term using colon ":"
```

Why do we say that the variables x_1 and x_3 ”interact”?:

If a non-interacting variable x_m increases by an amount of Δ, the response y increases by $\beta_m \cdot \Delta$, independent of any other variable. For example, if x_2 increases by Δ, the response increases by $\Delta \cdot \beta_2$. However, if x_3 increases by Δ, the response increases by $(\beta_3 + \beta_4 \cdot x_1) \cdot \Delta$, i.e. the increase depends on the variable x_1.

Multiple Linear Regression with Interaction

a) Let's try the additive model first (without interaction):

```
lm1 = lm(y ~ x1 + x2 + x3, data = mdata)
# Coefficients:
# (Intercept)         x1         x2         x3
#   -54540.06   1049.80   -55.75   213.66 # doesn't work!
```

b) Model with interaction:

```
lm2 = lm(y ~ x1 + x2 + x3 + x1:x3, data = mdata)
# Coefficients:
# (Intercept)        x1         x2         x3    x1:x3
#      5.1748   1.9506   3.0172   0.9822   4.0003 # much better,
#                                                  not perfect
```

- In practice, the correct interaction terms might not be known
- → dig up an appropiate model by trial and error
- "`add1`" or "`drop1`": add / remove terms step by step.
- Compare models using: `anova(lm1, lm2, test = "Chisq")`

Comparing Regression Models with ANOVA*

- In general, ANOVA compares variances
- → compare the residual variances of two regression models:
  - Model "Big": p_1 coefficients β_i
  - Model "Small": p_2 coefficients β_i , $p_2 < p_1$ (nested!)
- The bigger model will **always** be able to fit the data at least as well as the small model.
- But does "Big" give a **significantly better** fit to the data ?
  - → F test (used by ANOVA)
- H_0: "Big" does **not** give a significantly better fit than "Small"

If the null hypothesis is true, then:

$$F = \frac{\frac{SS_{res}^1 - SS_{res}^2}{p_2 - p_1}}{\frac{SS_{res}^2}{n - p_2}} \sim F(p_2 - p_1, n - p_2)$$

recall that:

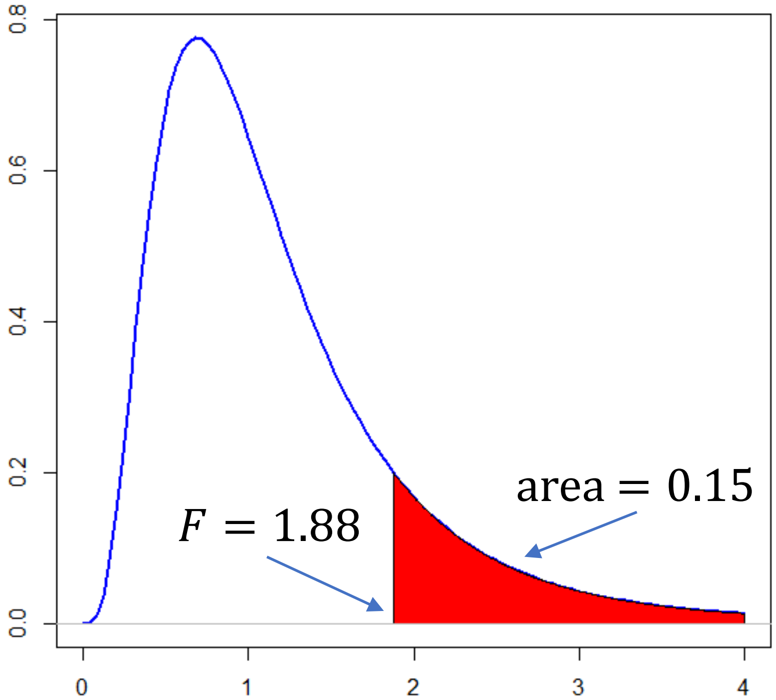
$$\frac{1}{\sigma^2} SS_{res} \sim \chi^2(f)$$

$$\frac{\frac{\chi^2(n)}{n}}{\frac{\chi^2(m)}{m}} \sim F(m, n)$$

A big value of the F-statistic would mean that there is a big difference between the sums of squares of both models. In that case, the null hypothesis is rejected.

Comparing Regression Models with ANOVA*

$$F = \frac{\frac{SS^1_{res} - SS^2_{res}}{p_2 - p_1}}{\frac{SS^2_{res}}{n - p_2}} \sim F(p_2 - p_1, n - p_2) \qquad \text{under } H_0$$



- here:
- $F = 1.88$ (observed)
- $p = 0.15$
- H_0 not rejected
- both models perform equally
- → choose the smaller model

Comparing Regression Models with ANOVA*

Another example:

```
anova(lm1, lm2, test = "Chisq") # comparison of nested lm1 and lm2
# Analysis of Variance Table
#
# Model 1: y ~ x1 + x2 + x3
# Model 2: y ~ x1 + x2 + x3 + x1:x3
#   Res.Df        RSS    Df    Sum of Sq           F      Pr(>F)
# 1      6  806403914
# 2      5         14    1    806403899  281359883  < 2.2e-16 ***
```

- Model 2 (with interaction) is significantly better ($p < 2.2e-16$)
- the better model has much lower lower Residual Sum of Squares (RSS)
- For the comparison to work, the models must be nested !
  - (the bigger model must include all terms of the smaller one)
- Find smallest model yielding "good" fit: Use additional predictors only if RSS is significantly reduced.

Automated Model Search

Aim: Find the smallest model which is "good enough“ which means that there is no bigger model which is **significantly** better

```
reduced = step(lm2, direction = "backward")          # shorten model stepwise
```

In this case, no smaller model was found (all coefficients still in "summary"):

```
summary(reduced)

# Coefficients:
#                  Estimate   Std. Error      t value     Pr(>|t|)
# (Intercept)     9.9565797   6.3879861        1.559        0.18
# x1              1.9762362   0.0703385       28.096    1.07e-06  ***
# x2              3.0086989   0.0137903      218.175    3.84e-11  ***
# x3              0.9763001   0.0188445       51.808    5.07e-08  ***
# x1:x3           3.9998565   0.0002609    15328.127     < 2e-16  ***
```

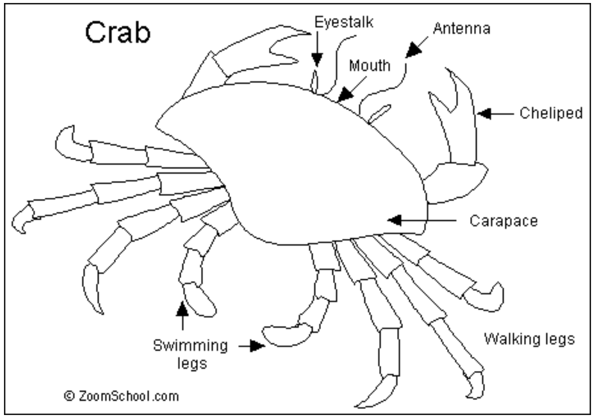
All p-values (except for the one corresponding to the intercept, which is of minor importance) are small, i.e. all corresponding coefficients $(\beta_1, \beta_2, \beta_3, \beta_4)$ are significantly different from zero. Hence, the response is actually depending on these variables, and the interaction term is necessary.

Multiple Regression Models with Categorical predictors

- **Categorical variables:** male/female ; smoking: yes/no ; risk: high/middle/low
- ANOVA: **all** explanatory variables are categorical
- Multiple Regression: explanatory variables can be continuous and/or categorical

Example from: http://www.utdallas.edu/~ammann/stat6338/node7.html
see General_Reg_Models_Examples.R

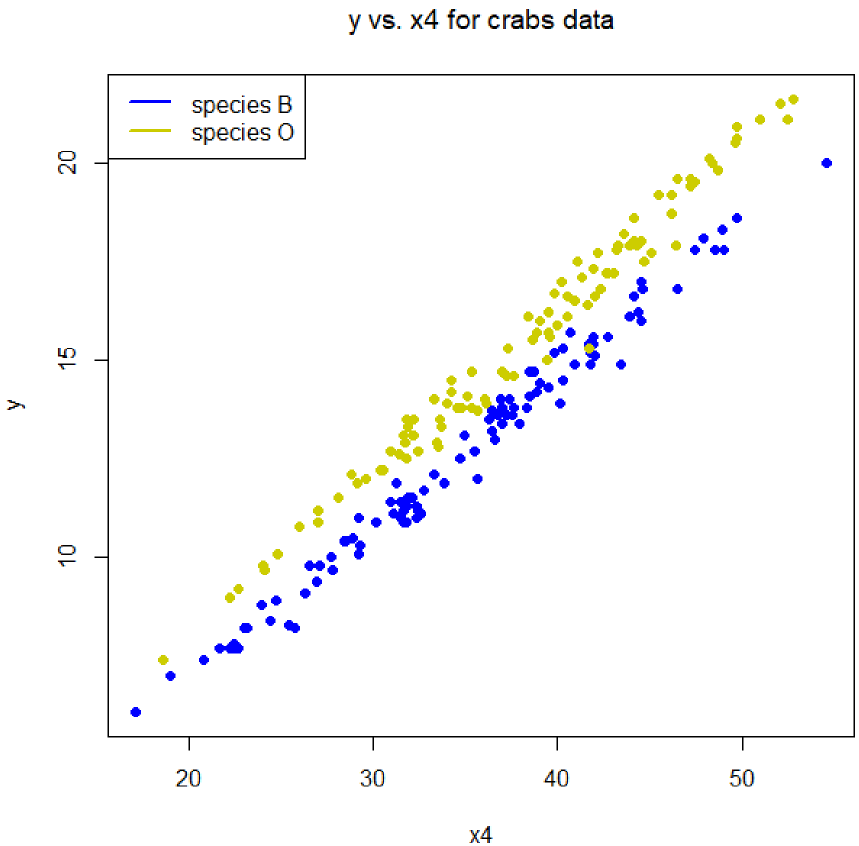
```
crabs = read.csv(file="crabs.csv", header=T)
head(crabs)

#    Species  Gender   x1    x2    x3    x4    y    # categ. & continuous predictors
# 1        B       M  8.1   6.7  16.1  19.0  7.0   # y is the response
# 2        B       M  8.8   7.7  18.1  20.8  7.4
# 3        B       M  9.2   7.8  19.0  22.4  7.7
# 4        B       M  9.6   7.9  20.1  23.1  8.2
# 5        B       M  9.8   8.0  20.3  23.0  8.2
# 6        B       M 10.8   9.0  23.0  26.5  9.8

levels(crabs$Species)    # "B" "O"
levels(crabs$Gender)     #  "F" "M"
```



Multiple Regression Models with Categorical predictors

Consider y versus x_4 for the different species (we ignore dependence on other variables for now):

```
  plot(y ~ x4, data=crabs[which(crabs$Species == "B"),], col="blue"...)
points(y ~ x4, data=crabs[which(crabs$Species == "O"),], col="yellow3", …)
```



Trends for the species fairly parallel … i.e species "O" adds some amount to the response **independently** of x_4.
→ probably **no interaction** between "Species" and x_4 → additive model

Multiple Regression Models with Categorical predictors

```
lm.a = lm(y ~ x4 + Species, data=crabs) # Additive, categorical & continuous

coef(lm.a)
#      (Intercept)          x4    SpeciesO
#       -1.3001043   0.3998935   1.5373614

beta0 = coef(lm.a)[1] # -1.3001043
beta1 = coef(lm.a)[2] # 0.3998935
beta2 = coef(lm.a)[3] # 1.537361
```

The W-R notation used above translates to the model:

$$y = \beta_0 + \beta_1 \cdot x_4 + \beta_2 \cdot I_{species} + \varepsilon$$

"Species" is a categorical variable $\rightarrow$ associated with indicator variable $I_{species}$:

$$I_{species} = \begin{cases} 0 & Species = B \\ 1 & Species = O \end{cases}$$

"B" = "base level" or "reference level", associated with the indicator value 0

Multiple Regression Models with Categorical predictors

$$y = \beta_0 + \beta_1 \cdot x_4 + \beta_2 \cdot I_{species} + \varepsilon$$

Calculation of slope / intercept for species "B" and "O" (when plotting y vs. x_4):

Species	Indicator	Model	Slope	Intercept
B	0	$y = \beta_0 + \beta_1 x_4 + \varepsilon$	β_1	β_0
O	1	$y = \beta_0 + \beta_1 x_4 + \beta_2 + \varepsilon$	β_1	$\beta_0 + \beta_2$

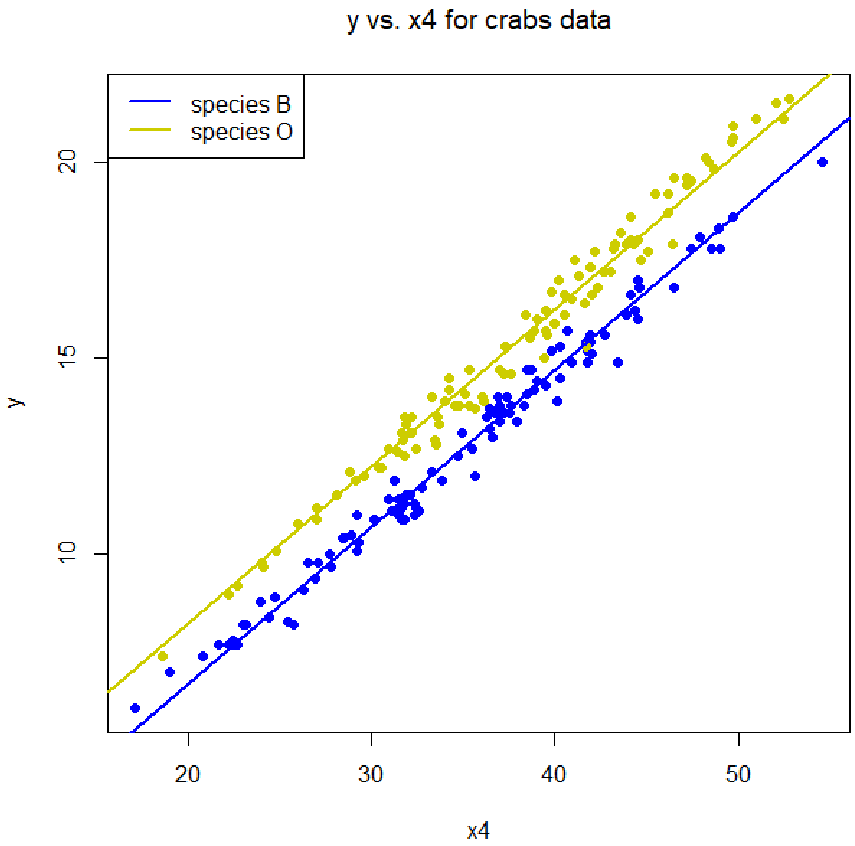
We decided to go for a model without interaction between x_4 and *species*. As a result, both regression lines have the same slope, so that they are parallel. Going from species "B" to species "O" adds the amount β_2 to the response independently of x_4. Let us add the regression lines for both species to the data:

```
abline(a = beta0 + beta2,  b = beta1, col = "yellow3", lty = 1, lwd = 2)
abline(a = beta0, b = beta1, col = "blue", lty = 1, lwd = 2) # same slope a
```

see General_Reg_Models_Examples.R

Multiple Regression Models with Categorical predictors

```
abline(a = beta0 + beta2,  b = beta1, col = "yellow3", lty = 1, lwd = 2)
abline(a = beta0, b = beta1, col = "blue", lty = 1, lwd = 2) # same slope a
```

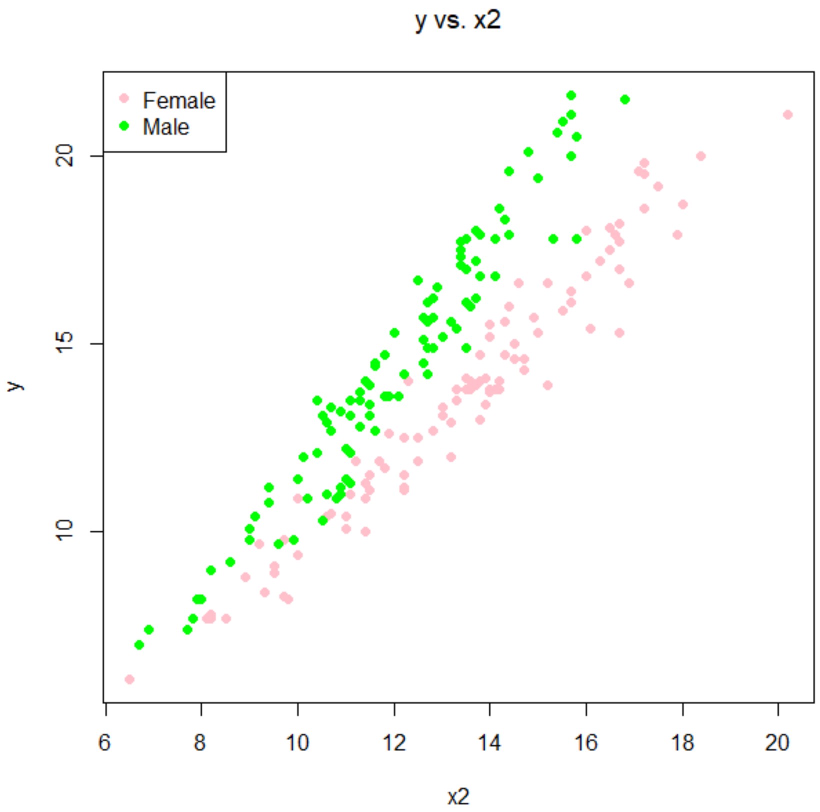


Changing from species "B" to species "O" (in terms of changing the object of attention) adds $\beta_2 = 1.54$ to the regression line.

Multiple Regression Models with Categorical predictors and Interaction

Now, consider the dependence of y from x_2 for the different genders:

```
  plot(y ~ x2, data=crabs[which(crabs$Gender == "F"),], col="pink", ..
points(y ~ x2, data=crabs[which(crabs$Gender == "M"),], col="green",..
```



Trends for "Female" and "Male" seem to have different slopes
For higer x_2, the Gender effect is more pronounced
→ a regression Model including an interaction terms is advisable

Multiple Regression Models with Categorical predictors and Interaction

a) Try **without interaction** first:

```
lm.0 = lm(y ~ x2 + Gender, data=crabs) # additive model
summary(lm.0) # (shortened)

#               Estimate  Std. Error   t value    Pr(>|t|)
# (Intercept)  -4.23317     0.38106    -11.11      <2e-16 ***
# x2            1.33144     0.02736     48.66      <2e-16 ***
# GenderM       2.60617     0.14047     18.55      <2e-16 *** # baselevel
```

According to Wilkinson-Rogers notation, `y ~ x2 + Gender` translates to

$$y = \beta_0 + \beta_1 \cdot x_2 + \beta_2 \cdot I_{gender} + \varepsilon$$

$$I_{gender} = \begin{cases} 0 & Gender = Female \\ 1 & Gender = Male \end{cases}$$

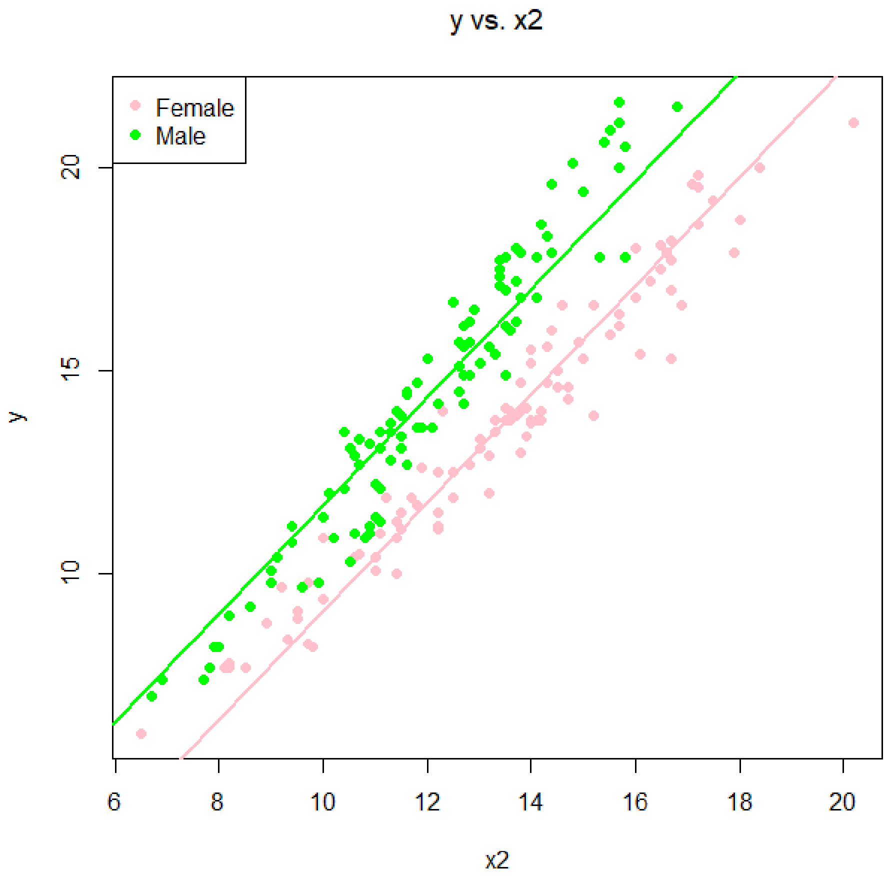
Gender	Indicator	Model	Slope	Intercept
F	0	$y = \beta_0 + \beta_1 x_2 + \varepsilon$	β_1	β_0
M	1	$y = \beta_0 + \beta_1 x_2 + \beta_2 + \varepsilon$	β_1	$\beta_0 + \beta_2$

Multiple Regression Models with Categorical predictors and Interaction

```
lm.0 = lm(y ~ x2 + Gender, data = crabs)    # additive model
summary(lm.0)

beta0 = coef(lm.0)[1]      # -4.233172
beta1 = coef(lm.0)[2]      # 1.331443
beta2 = coef(lm.0)[3]      # 2.60617

slope.female =  beta1
icept.female =  beta0
slope.male = beta1         # same slope as female
icept.male = beta0 + beta2

abline(a=icept.female, b=slope.female, col="pink",  lty=1, lwd=2)
abline(a=icept.male,   b=slope.male,   col="green", lty=1, lwd=2)
```

Multiple Regression Models with Categorical predictors and Interaction

```
abline(a=icept.female, b=slope.female, col="pink",  lty=1, lwd=2)
abline(a=icept.male,   b=slope.male,   col="green", lty=1, lwd=2)
```



It seems that a model yielding the same slope for both datasets (Female, Male) does not work → interaction term needed

Multiple Regression Models with Categorical predictors and Interaction

b) Model **with interaction**:

```
lm.b = lm(y ~ x2*Gender, data=crabs) # with interaction
summary(lm.b)   # (shortened)

#                Estimate  Std. Error  t value  Pr(>|t|)
# (Intercept)  -2.29012     0.42271   -5.418  1.76e-07 ***
# x2            1.18737     0.03072   38.651   < 2e-16 ***
# GenderM      -2.11660     0.63564   -3.330   0.00104 **
# x2:GenderM    0.37590     0.04962    7.575  1.38e-12 ***
```

According to Wilkinson-Rogers notation, `y ~ x2*Gender` translates to

$$y = \beta_0 + \beta_1 \cdot x_2 + \beta_2 \cdot I_{gender} + \underline{\beta_3 \cdot x_2 \cdot I_{gender}} + \varepsilon$$

$$I_{gender} = \begin{cases} 0 & Gender = Female \\ 1 & Gender = Male \end{cases}$$

Multiple Regression Models with Categorical predictors and Interaction

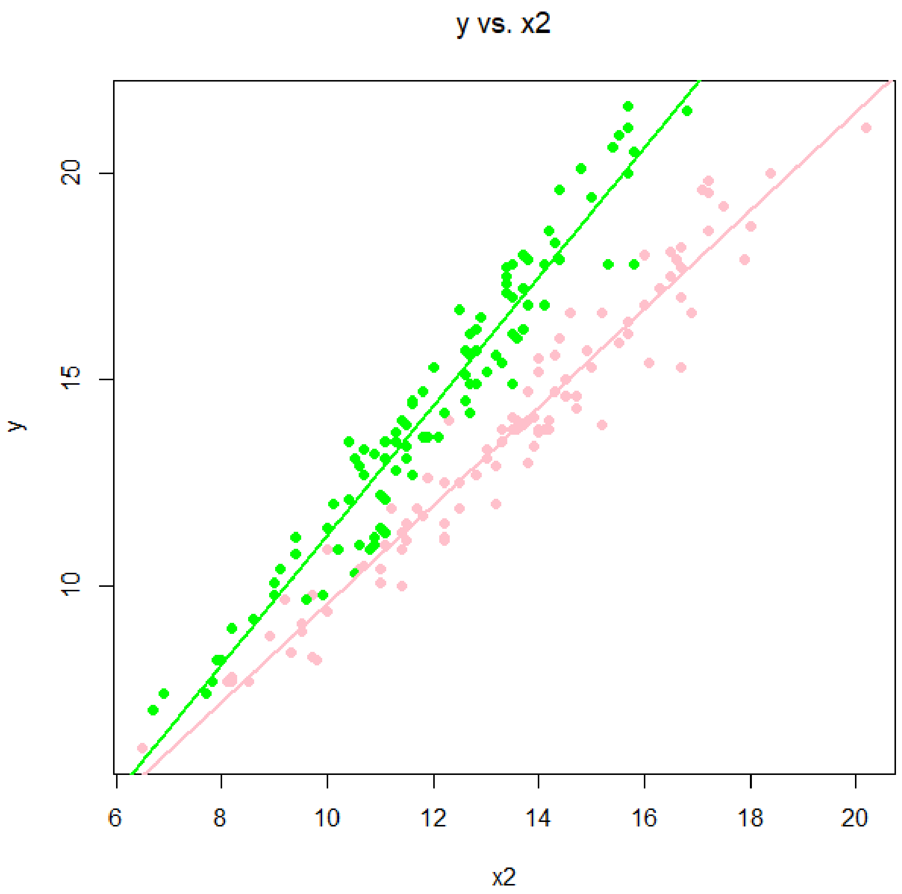
$$y = \beta_0 + \beta_1 \cdot x_2 + \beta_2 \cdot I_{gender} + \beta_3 \cdot x_2 \cdot I_{gender} + \varepsilon$$

Gender	I	Model	Slope	Intercept
F	0	$y = \beta_0 + \beta_1 x_2 + \varepsilon$	β_1	β_0
M	1	$y = \beta_0 + \beta_1 x_2 + \beta_2 + \beta_3 x_2 + \varepsilon$	$\beta_1 + \beta_3$	$\beta_0 + \beta_2$

This is a model yielding different slopes and intercepts for both genders.

```
lm.b = lm(y ~ x2*Gender, data=crabs) # with interaction
beta0 = coef(lm.b)[1]
beta1 = coef(lm.b)[2]
beta2 = coef(lm.b)[3]
beta3 = coef(lm.b)[4]
slope.female = beta1
icept.female = beta0
slope.male = beta1 + beta3
icept.male = beta0 + beta2
plot(y ~ x2, data=crabs[which(crabs$Gender == "F"),], col="pink", …
points(y ~ x2, data=crabs[which(crabs$Gender == "M"),], col="green", …)
abline(a=icept.female, b=slope.female, col="pink",  lty=1, lwd=2)
abline(a=icept.male,   b=slope.male,   col="green", lty=1, lwd=2)
```

Multiple Regression Models with Categorical predictors and Interaction



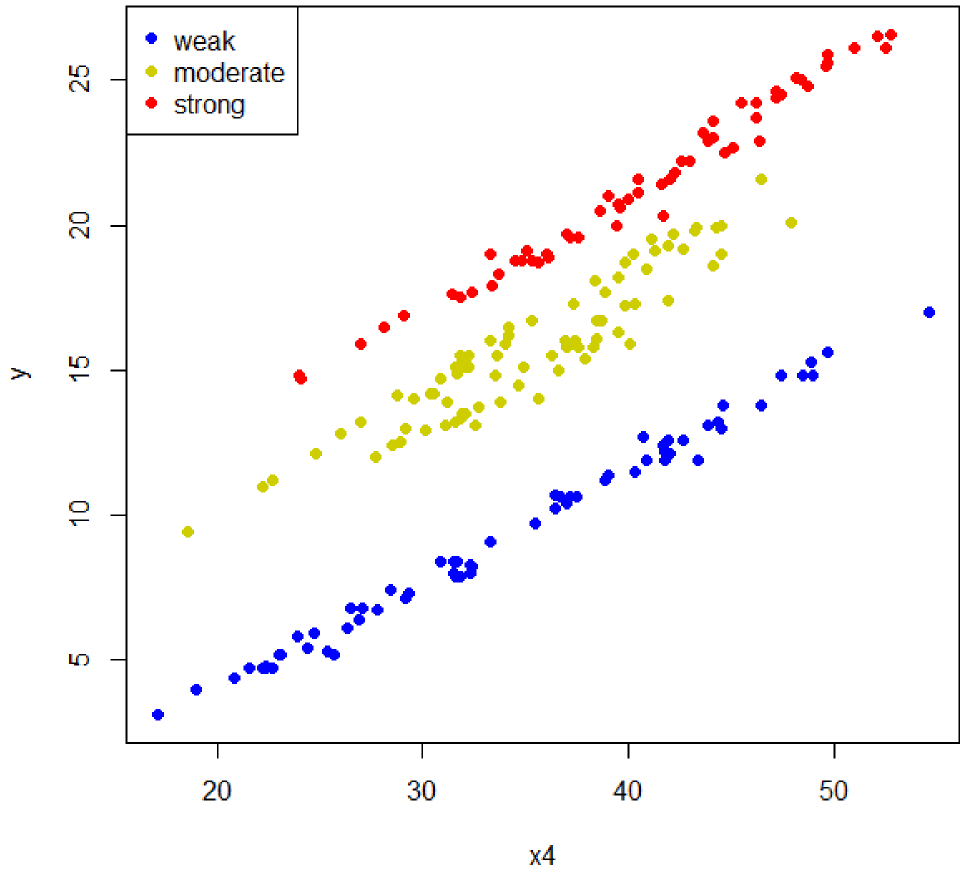
- A model including interaction provides a better fit.
- The regression lines for different genders have different slopes.

Multiple Regression Models with Multilevel Categorical Predictors

- The factor "Gender" considered above had two levels: female / male → we needed one indicator variable I_{gender} to build a regression model
- **In general**: Factors with L levels require $L-1$ indicator variables.
- Let us look at a categorical variable with 3 levels:

```
effect = read.csv(file = "Effects.csv", header = T)
levels(effect$effect)  # "moderate"  "strong"   "weak"

  plot(y ~ x4, data=effect[which(effect$effect == "weak"),],….
points(y ~ x4, data=effect[which(effect$effect == "moderate"),], ….
points(y ~ x4, data=effect[which(effect$effect == "strong"),], ….
```

Multiple Regression Models with Multilevel Categorical Predictors



The trends for the different levels are fairly parallel → no interaction between the categorical variable "effect" and the continuous variable x_4 → use additive model

Multiple Regression Models with Multilevel Categorical Predictors

No interaction, use additive model:

```
fit <- lm(y ~ x4 + effect, data = effect)  # additive
summary(fit)  # shortened
# Coefficients:
#                Estimate Std. Error t value Pr(>|t|)
# (Intercept)    1.624953   0.232370   6.993 4.14e-11 ***
# x4             0.400579   0.006268  63.908  < 2e-16 ***
# effectstrong   3.462941   0.118086  29.326  < 2e-16 ***
# effectweak    -6.001887   0.110700 -54.217  < 2e-16 ***
```

According to Wilkinson-Rogers notation, `y ~ x4 + effect` translates to

$$y = \beta_0 + \beta_1 \cdot x_4 + \beta_2 \cdot I_1 + \beta_3 \cdot I_2 + \varepsilon$$

$$I_1 = \begin{cases} 0 & effect = moderate \\ 1 & effect = strong \\ 0 & effect = weak \end{cases} \qquad I_2 = \begin{cases} 0 & effect = moderate \\ 0 & effect = strong \\ 1 & effect = weak \end{cases}$$

The level `moderate` is chosen as base level because it comes first in the alphabet (the command `levels()` lists the base level first)

Multiple Regression Models with Multilevel Categorical Predictors

$$y = \beta_0 + \beta_1 \cdot x_4 + \beta_2 \cdot I_1 + \beta_3 \cdot I_2 + \varepsilon$$

$$I_1 = \begin{cases} 0 & effect = moderate \\ 1 & effect = strong \\ 0 & effect = weak \end{cases} \qquad I_2 = \begin{cases} 0 & effect = moderate \\ 0 & effect = strong \\ 1 & effect = weak \end{cases}$$

Effect	I_1	I_2	Model	Slope	Intercept
moderate	0	0	$y = \beta_0 + \beta_1 x_4 + \varepsilon$	β_1	β_0
strong	1	0	$y = \beta_0 + \beta_1 x_4 + \beta_2 + \varepsilon$	β_1	$\beta_0 + \beta_2$
weak	0	1	$y = \beta_0 + \beta_1 x_4 + \beta_3 + \varepsilon$	β_1	$\beta_0 + \beta_3$

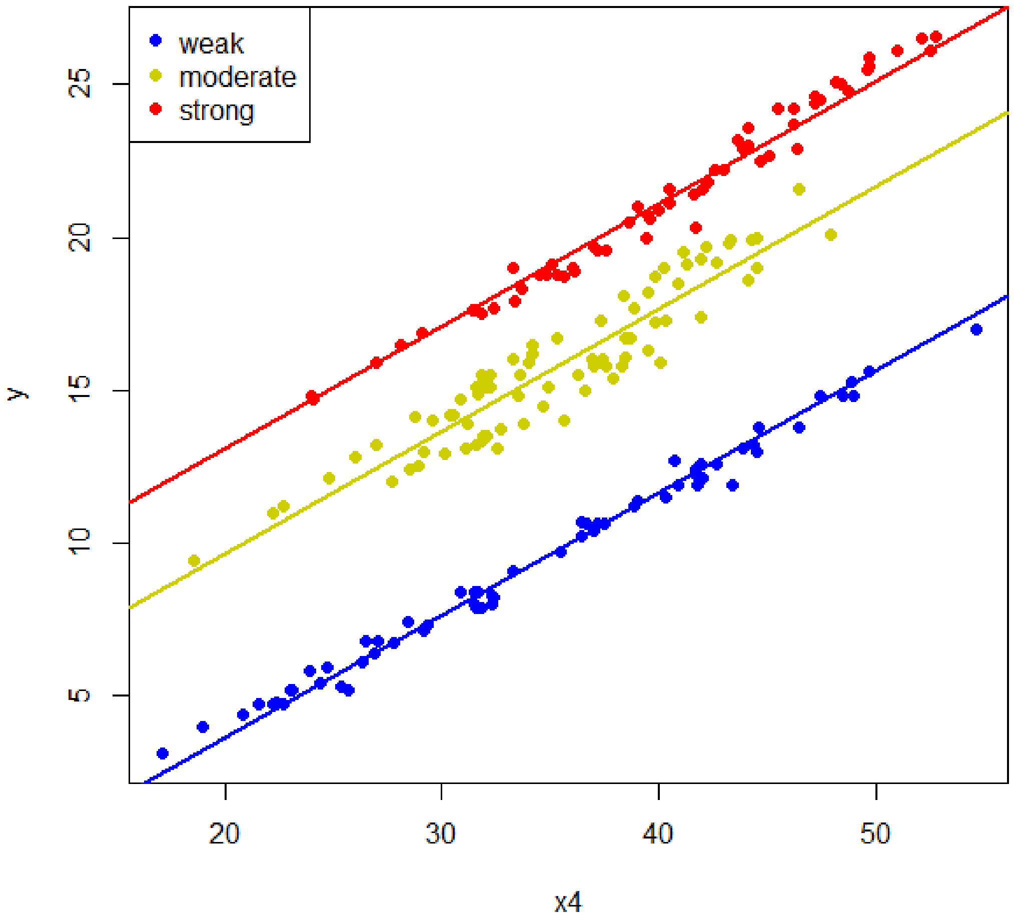
Levels are ordered according to the alphabet. The level ”moderate” is the base level, both indicators are assigned a zero value. The level ”strong” is connected with value 1 for I_1, ”weak” is connected with value 1 for I_2.

Multiple Regression Models with Multilevel Categorical Predictors

Effect	I_1	I_2	Model	Slope	Intercept
moderate	0	0	$y = \beta_0 + \beta_1 x_4 + \varepsilon$	β_1	β_0
strong	1	0	$y = \beta_0 + \beta_1 x_4 + \beta_2 + \varepsilon$	β_1	$\beta_0 + \beta_2$
weak	0	1	$y = \beta_0 + \beta_1 x_4 + \beta_3 + \varepsilon$	β_1	$\beta_0 + \beta_3$

```
coef(fit)
# (Intercept)            x4 effectstrong   effectweak
#   1.6249534     0.4005791     3.4629406   -6.0018875

beta0 = coef(fit)[1] # 1.624953   (Intercept)
beta1 = coef(fit)[2] # 0.4005791   x4
beta2 = coef(fit)[3] # 3.462941   effectstrong
beta3 = coef(fit)[4] # -6.001887  effectweak

slope.moderate = beta1
inter.moderate = beta0
slope.strong = beta1
inter.strong = beta0 + beta2
slope.weak = beta1
inter.weak = beta0 + beta3

abline(a = inter.moderate, b = slope.moderate, col = "yellow3", lty = 1, lwd = 2)
abline(a = inter.strong, b = slope.strong, col = "red", lty = 1, lwd = 2)
abline(a = inter.weak, b = slope.weak, col = "blue", lty = 1, lwd = 2)
```

Multiple Regression Models with Multilevel Categorical Predictors



Multiple Regression Models

- Comparison of models -

Show that `lm.b` (with interaction) is better than `lm.0` (no interaction):

```
anova(lm.0, lm.b, test="Chisq") # F-statistic = ratio of two chi^2

# Analysis of Variance Table
#
# Model 1: y ~ x2 + Gender
# Model 2: y ~ x2 * Gender
#   Res.Df    RSS  Df   Sum of Sq    Pr(>Chi)
# 1    197 177.83
# 2    196 137.55   1      40.272  3.586e-14 ***
```

The p-value indicates that there is a significant difference between the performance of the two models. Model 2 (with interaction) is the better model - the residual sum of squares (RSS) is lower.