Introduction to Linear Regression

Arnaud Legrand and Jean-Marc Vincent

Scientific Methodology and Performance Evaluation

Outline

What is a regression?

Regression analysis is the most widely used statistical tool for understanding relationships among variables. Several possible objectives including:

1. Prediction of future observations. This includes extrapolation since we all like connecting points by lines when we *expect* things to be continuous
2. Assessment of the effect of, or relationship between, explanatory variables on the response
3. A general description of data structure (generally expressed in the form of an equation or a model connecting the response or dependent variable and one or more explanatory or predictor variable)
4. Defining what you should "expect" as it allows you to define and detect what does not behave as expected

The linear relationship is the most commonly found one

- we will illustrate how it works
- it is very general and is the basis of many more advanced tools (polynomial regression, ANOVA, ...)

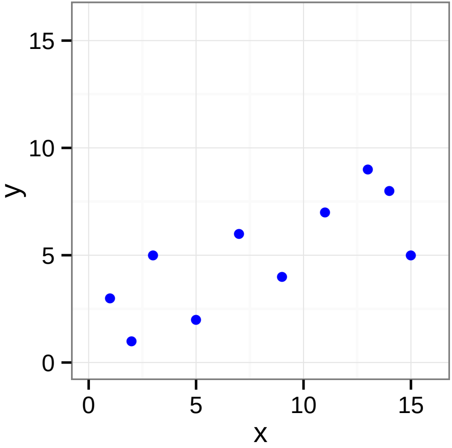
Outline

Starting With a Simple Data Set

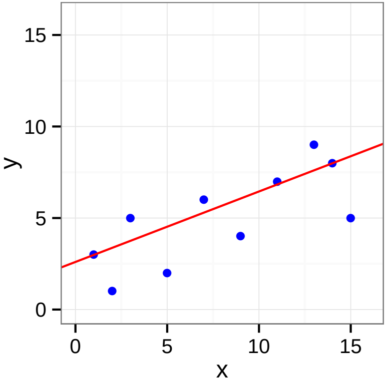
Descriptive statistics provides simple summaries about the sample and about the observations that have been made.

How could we summarize the following data set ?

	x	y
1	1.00	3.00
2	2.00	1.00
3	3.00	5.00
4	5.00	2.00
5	7.00	6.00
6	9.00	4.00
7	11.00	7.00
8	13.00	9.00
9	14.00	8.00
10	15.00	5.00

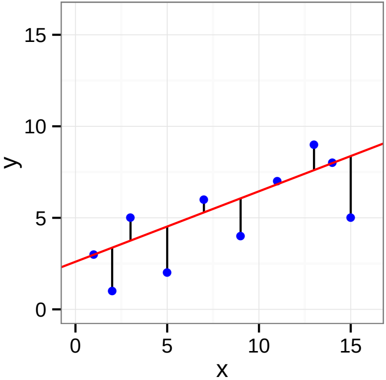


The "Eyeball" Method



- A straight line drawn through the maximum number of points on a scatter plot balancing about an equal number of points above and below the line
- Some points are rather far from the line. Maybe we should instead try to minimize some kind of *distance to the line*

Least Squares Line (1): What to minimize?



Intuitively, a large error is *much* more important than a small one. We could try to minimize $F(\alpha, \beta) = \sum_i \underbrace{(y_i - \alpha - \beta x_i)}_{e_i}{}^2$, the size of all residuals:

- If they were all zero we would have a perfect line
- Trade-off between moving closer to some points and at the same time moving away from other points

Least Squares Line (2): Simple Formula

$$F(\alpha, \beta) = \sum_{i=1}^{n} (y_i - \alpha - \beta x_i)^2$$

F is quadratic in α and in β so if we simply differentiate F by α and by β, we can obtain a closed form for the minimum:

$$\hat{\beta} = \frac{\sum_{i=1}^{n}(x_i - \bar{x})(y_i - \bar{y})}{\sum_{i=1}^{n}(x_i - \bar{x})^2} = \frac{\sum_{i=1}^{n} x_i y_i - \frac{1}{n}\sum_{i=1}^{n} x_i \sum_{j=1}^{n} y_j}{\sum_{i=1}^{n}(x_i^2) - \frac{1}{n}(\sum_{i=1}^{n} x_i)^2}$$

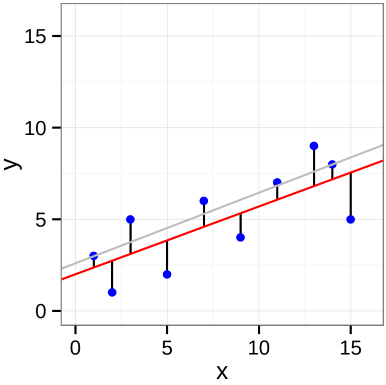
$$= \frac{\overline{xy} - \bar{x}\bar{y}}{\overline{x^2} - \bar{x}^2} = \frac{\mathrm{Cov}[x, y]}{\mathrm{Var}[x]} = r_{xy}\frac{s_y}{s_x}$$

$\hat{\alpha} = \bar{y} - \hat{\beta}\,\bar{x}$, where:

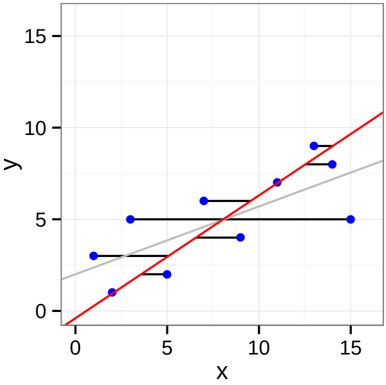
- $\bar{x}$ and $\bar{y}$ are the sample mean of x and y
- r_{xy} is the sample correlation coefficient between x and y
- s_x and s_y are the sample standard deviation of x and y

Also it has a good geometric interpretation (orthogonal projection)

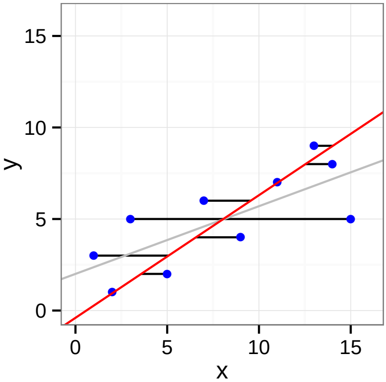
Least Squares Line (3): y as a function of x or the opposite?



Least Squares Line (3): y as a function of x or the opposite?

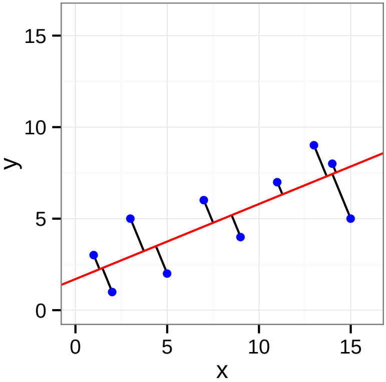


Least Squares Line (3): y as a function of x or the opposite?



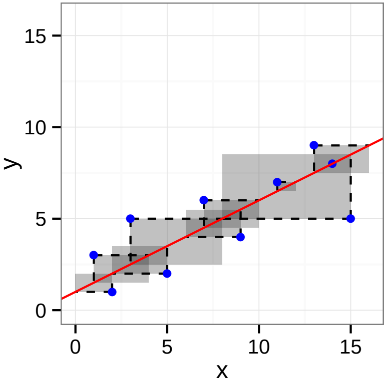
OK, do we have less asymetrical options?

Least Distances Line (a.k.a. Deming Regression)



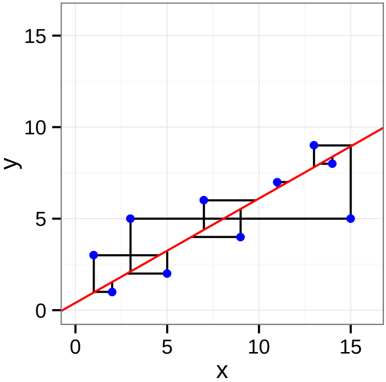
- Note that somehow, this makes sense only if we have a square plot, i.e., if x and y have the same units

Least Rectangles Line



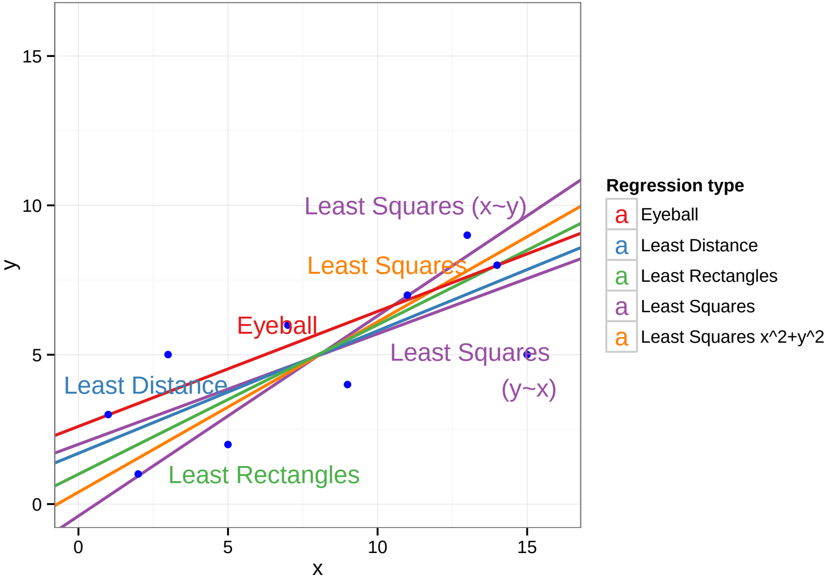
- Minimize $E(\alpha, \beta) = \sum_{i=1}^{n} \left| x_i - \frac{y_i - \alpha}{\beta} \right| \cdot |y_i - \alpha - \beta x_i|$
- This leads to the regression line $y = \frac{s_y}{s_x}(x - \bar{x}) + \bar{y}$.

Least Squares (in Both Directions) Line



- Minimize $D(\alpha, \beta) = \sum_{i=1}^{n} \left(x_i - \frac{y_i - \alpha}{\beta}\right)^2 + (y_i - \alpha - \beta x_i)^2$
- Has to be computed analytically

Which line to choose?



What does correspond to each line?

- Eyeball: AFAIK nothing
- Least Squares: classical linear regression $y \sim x$
- Least Squares in both directions: I don't know
- Deming: equivalent to Principal Component Analysis
- Rectangles: may be used when one variable is not "explained" by the other, but are inter-dependent

This is not just a geometric problem. You need a model of to decide which one to use

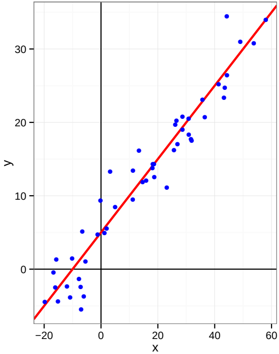
Outline

The Simple Linear Regression Model

We need to invest in a probability model

$$Y = a + bX + \varepsilon$$

- Y is the response variable
- X is a continuous explanatory variable
- a is the intercept
- b is the slope
- ε is some noise



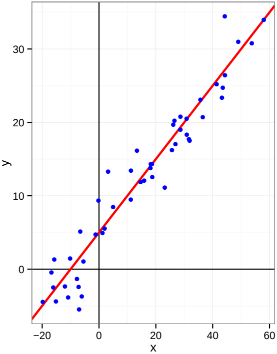
- $a + bX$ represents the "true line", the part of Y that depends on X
- The error term ε is independent "idosyncratic noise", i.e., the part of Y not associated with X

The Simple Linear Regression Model

We need to invest in a probability model

$$Y = a + bX + \varepsilon$$

- Y is the response variable
- X is a continuous explanatory variable
- a is the intercept
- b is the slope
- ε is some noise



- $a + bX$ represents the "true line", the part of Y that depends on X
- The error term ε is independent "idosyncratic noise", i.e., the part of Y not associated with X

Gauss-Markov Theorem

Under a few assumptions, the least squares regression is the Best Linear Unbiased Estimate

- $\mathrm{E}[\hat{\beta}] = b$ and $\mathrm{E}[\hat{\alpha}] = a$
- $\mathrm{Var}(\hat{\beta})$ and $\mathrm{Var}(\hat{\alpha})$ are minimal

Multiple explanatory variables

- The same results hold true when there are several explanatory variables:
$$Y = a + b^{(1)}X^{(1)} + b^{(2)}X^{(2)} + b^{(1,2)}X^{(1)}X^{(2)} + \varepsilon$$
The least squares regressions are good estimators of a, $b^{(1)}$, $b^{(2)}$, $b^{(1,2)}$

- We can use an arbitrary linear combination of variables, hence
$$Y = a + b^{(1)}X + b^{(2)}\frac{1}{X} + b^{(3)}X^3 + \varepsilon$$
is also a linear model
- Obviously the closed-form formula are much more complicated but softwares like R handle this very well

Outline

Important Hypothesis (1)

Weak exogeneity The predictor variables X can be treated as fixed values, rather than random variables: the X are assumed to be error-free, i.e., they are not contaminated with measurement errors

Although not realistic in many settings, dropping this assumption leads to significantly more difficult *errors-in-variables* models

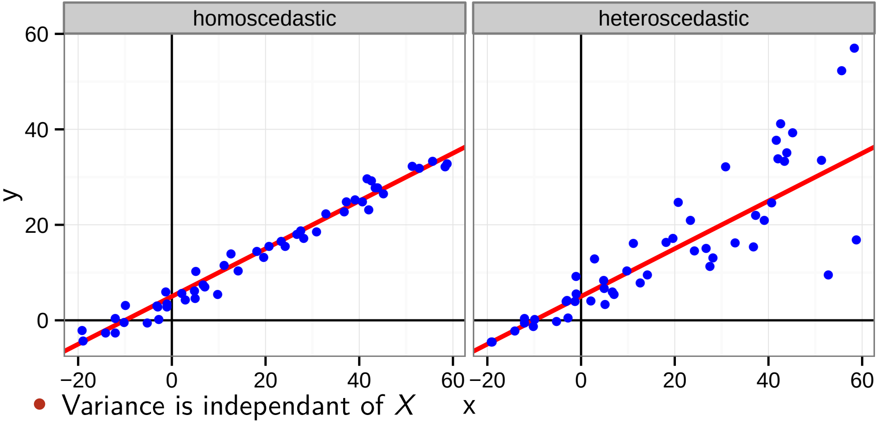
Linearity the mean of the response variable is a linear combination of the parameters (regression coefficients) and the predictor variables

Since predictor variables themselves can be arbitrarily transformed, this is not that restrictive. This trick is used, for example, in polynomial regression, but beware of overfitting

Independance of Errors if several responses Y_1 and Y_2 are fit, ε_1 and ε_2 should be independant

Other Very Important Hypothesis

Constant variance (a.k.a. homoscedasticity)

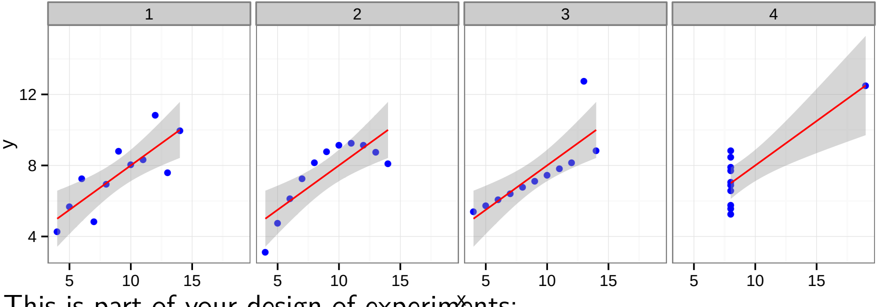


- Variance is independant of X
- If several responses Y_1 and Y_2 are fit, ε_1 and ε_2 should have the same variance
- Either normalize Y or use an other estimator

Other Classical Hypothesis (3)

Normal and iid errors This is not an assumption of the Gauss Markov Theorem. Yet, it is quite convenient to build *precise* confidence intervals of the regression

Arrangement of the predictor variables X it has a major influence on the precision of estimates of β (remember Anscombe's quartet).

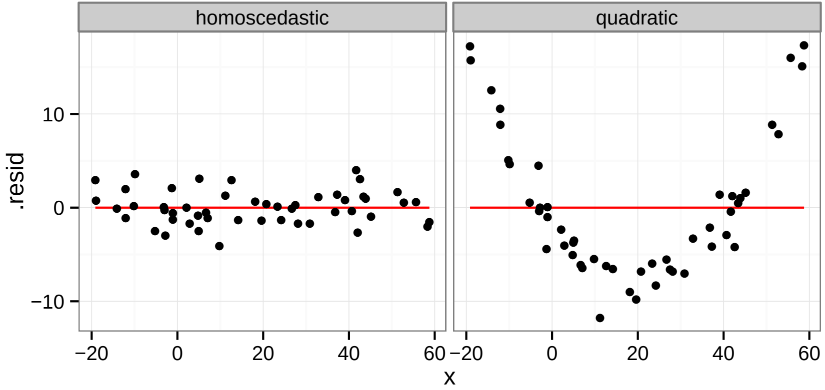


This is part of your design of experiments:

- If you want to test linearity, X should be uniformly distributed
- If you want the best estimation, you should use extreme values of X

Outline

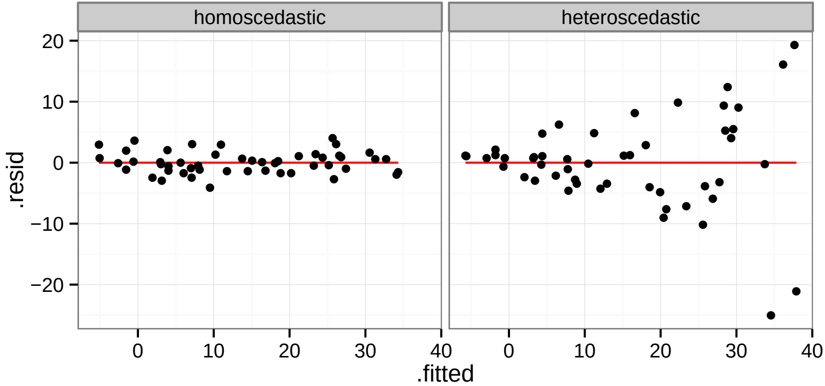
Linearity: Residuals vs. Explanatory Variable



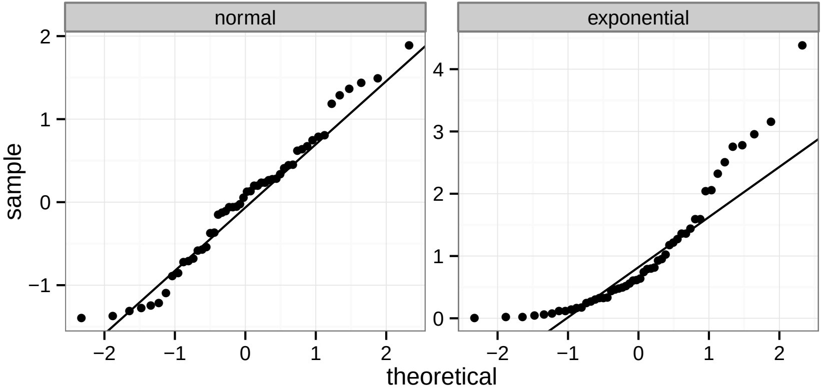
When there are several factors, you have to check for every dimension...

⌢̈ That's why one generally study residuals as a function of fitted values.

Homoscedasticity: Residuals vs. Fitted values



Normality: qqplots



A quantile-quantile plot is a graphical method for comparing two probability distributions by plotting their quantiles against each other

Model Formulae in R

The structure of a model is specified in the formula like this:

response variable ~ explanatory variable(s)

~ reads "is modeled as a function of " and `lm(y~x)` means $y = a + bx + \varepsilon$

On the right-hand side, one should specify how the explanatory variables are combined. The symbols used here have a different meaning than in arithmetic expressions

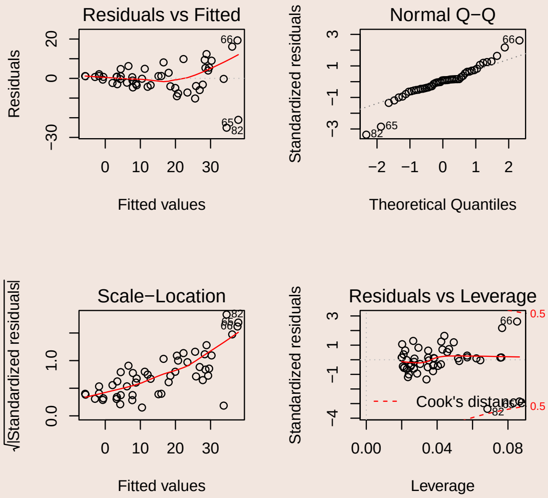
- `+` indicates a variable inclusion (not an addition)
- `-` indicates a variable deletion (not a substraction)
- `*` indicates inclusion of variables and their interactions
- `:` means an interaction

Therefore

- `z~x+y` means $z = a + b_1 x + b_2 y + \varepsilon$
- `z~x*y` means $z = \alpha + b_1 x + b_2 y + b_3 xy + \varepsilon$
- `z~(x+y)^2` means the same
- `log(y)~I(1/x)+x+I(x^2)` means $z = \alpha + b_1 \frac{1}{x} + b_2 x + b_3 x^2 + \varepsilon$

Checking the model with R

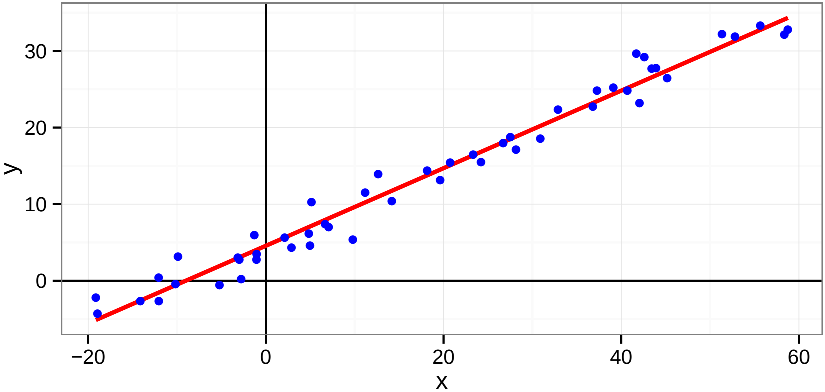
```r
reg <- lm(data=df[df$type=="heteroscedastic",],y~x)
par(mfrow=c(2,2));   plot(reg);   par(mfrow=c(1,1))
```



Outline

Decomposing the Variance

How well does the least squares line explain variation in Y?



Decomposing the Variance

How well does the least squares line explain variation in Y?

We have $Y = \hat{Y}(X) + \varepsilon$ $\quad$ ($\hat{Y}$ is the "true mean"; we note $\hat{Y} = \hat{Y}(X)$).
Since $\hat{Y}$ and ε are uncorrelated, we have

$$\mathrm{Var}(Y) = \mathrm{Var}(\hat{Y} + \varepsilon) = \mathrm{Var}(\hat{Y}) + \mathrm{Var}(\varepsilon)$$

$$\frac{1}{n-1}\sum_{i=1}^{n}(Y_i - \bar{Y})^2 = \frac{1}{n-1}\sum_{i=1}^{n}(\hat{Y}_i - \overline{\hat{Y}})^2 + \frac{1}{n-1}\sum_{i=1}^{n}(\varepsilon_i - \bar{\varepsilon})^2$$

Since $\bar{\varepsilon} = 0$ and $\bar{Y} = \overline{\hat{Y}}$, we have

$$\underbrace{\sum_{i=1}^{n}(Y_i - \bar{Y})^2}_{\text{Total Sum of Squares}} = \underbrace{\sum_{i=1}^{n}(\hat{Y}_i - \bar{Y})^2}_{\text{Regression SS}} + \underbrace{\sum_{i=1}^{n}\varepsilon_i{}^2}_{\text{Error SS}}$$

- SSR = Variation in Y explained by the regression line
- SSE = Variation in Y that is left unexplained

$SSR = SST \Rightarrow$ perfect fit

A Goodness of Fit Measure: R^2

The coefficient of determination, denoted by R^2, measures goodness of fit:

$$R^2 = 1 - \frac{SSE}{SST} = 1 - \frac{\sum_{i=1}^{n}(y_i - \hat{y}(x_i))^2}{\sum_{i=1}^{n}(y_i - \bar{y})^2} = 1 - \frac{\text{the error knowing } x}{\text{the error without knowing } x}$$

- $0 \leq R^2 \leq 1$
- The closer R^2 is to 1, the better the fit

Warning:

- A not so low R^2 may mean important noise or bad model
  - In biology or social sciences, an R^2 of .6 can be considered as good
  - In physics/engineering, an R^2 of .6 would be considered as low
- As you add parameters to a model, you inevitably improve the fit
  - The adjusted R^2 tries to compensate this
  - There is a trade-off beteween model simplicity and fit. Strive for simplicity!

Illustration with R (homoscedastic data)

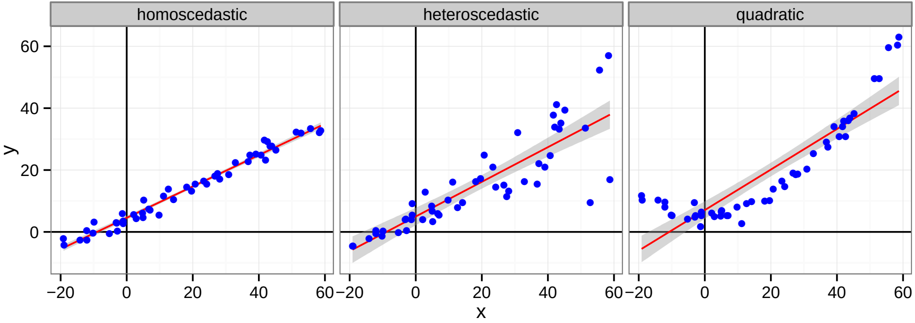


Illustration with R (homoscedastic data)

```
reg <- lm(data=df[df$type=="homoscedastic",],y~x)
summary(reg)
```

Illustration with R (homoscedastic data)

```r
reg <- lm(data=df[df$type=="homoscedastic",],y~x)
summary(reg)
```

```
Call:
lm(formula = y ~ x, data = df[df$type == "homoscedastic", ])

Residuals:
    Min      1Q  Median      3Q     Max
-4.1248 -1.3059 -0.0366  1.0588  3.9965

Coefficients:
            Estimate Std. Error t value Pr(>|t|)
(Intercept)  4.56481    0.33165   13.76   <2e-16 ***
x            0.50645    0.01154   43.89   <2e-16 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 1.816 on 48 degrees of freedom
Multiple R-squared:  0.9757,      Adjusted R-squared:  0.9752
F-statistic:  1926 on 1 and 48 DF,  p-value: < 2.2e-16
```

Illustration with R (homoscedastic data)

- `Std. Error` $= \sigma/\sqrt{n}$ and can be used are used to compute C.I on the regression estimates
- `t-value` and `Pr(>|t|)`: \$t\$-test whether $\mu \neq 0$
  - Easy to read significance codes
  - Assumes normality
- `F-statistic`: test the null hypothesis that all of the model coefficients are 0

```
Coefficients:
            Estimate Std. Error t value Pr(>|t|)
(Intercept)  4.56481    0.33165   13.76   <2e-16 ***
x            0.50645    0.01154   43.89   <2e-16 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 1.816 on 48 degrees of freedom
Multiple R-squared:  0.9757,     Adjusted R-squared:  0.9752
F-statistic:  1926 on 1 and 48 DF,  p-value: < 2.2e-16
```

Illustration with R (heteroscedastic data)

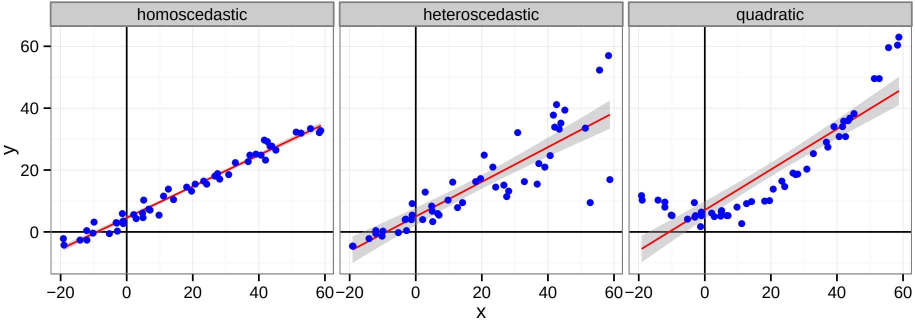


Illustration with R (heteroscedastic data)

```
reg <- lm(data=df[df$type=="heteroscedastic",],y~x)
summary(reg)
```

Illustration with R (heteroscedastic data)

```r
reg <- lm(data=df[df$type=="heteroscedastic",],y~x)
summary(reg)
```

```
Call:
lm(formula = y ~ x, data = df[df$type == "heteroscedastic", ])

Residuals:
    Min      1Q  Median      3Q     Max
-25.063  -3.472   0.663   3.707  19.327

Coefficients:
            Estimate Std. Error t value Pr(>|t|)
(Intercept)  4.98800    1.41061   3.536 0.000911 ***
x            0.56002    0.04908  11.411 2.83e-15 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 7.722 on 48 degrees of freedom
Multiple R-squared:  0.7306,	Adjusted R-squared:  0.725
F-statistic: 130.2 on 1 and 48 DF,  p-value: 2.83e-15
```

Illustration with R (quadratic data)

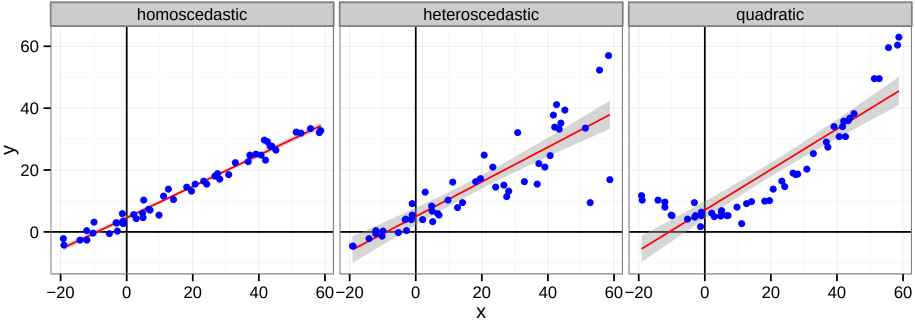


Illustration with R (quadratic data)

```r
reg <- lm(data=df[df$type=="quadratic",],y~x)
summary(reg)
```

Illustration with R (quadratic data)

```r
reg <- lm(data=df[df$type=="quadratic",],y~x)
summary(reg)
```

```
Call:
lm(formula = y ~ x, data = df[df$type == "quadratic", ])

Residuals:
    Min      1Q  Median      3Q     Max
-11.759  -5.847  -2.227   3.746  17.346

Coefficients:
            Estimate Std. Error t value Pr(>|t|)
(Intercept)  7.05330    1.41238   4.994 8.23e-06 ***
x            0.65517    0.04914  13.333  < 2e-16 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 7.732 on 48 degrees of freedom
Multiple R-squared:  0.7874,        Adjusted R-squared:  0.783
F-statistic: 177.8 on 1 and 48 DF,  p-value: < 2.2e-16
```

Illustration with R (quadratic data, polynomial regression)

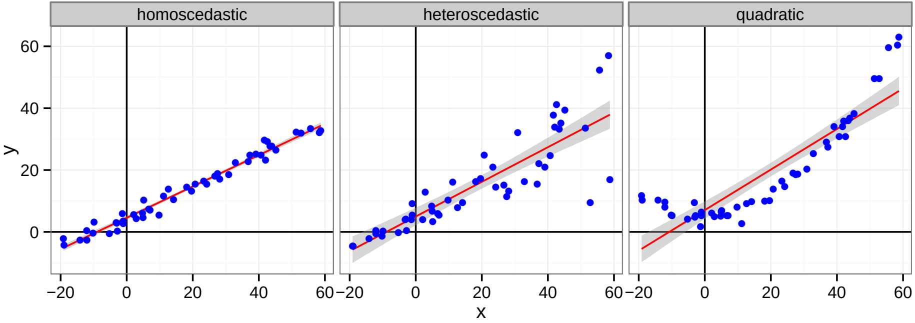


Illustration with R (quadratic data, polynomial regression)

```r
df$x2=df$x^2
reg_quad <- lm(data=df[df$type=="quadratic",],y~x+x2)
summary(reg_quad)
```

Illustration with R (quadratic data, polynomial regression)

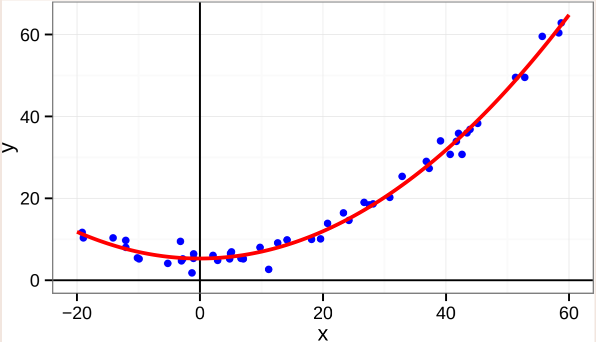
```r
df$x2=df$x^2
reg_quad <- lm(data=df[df$type=="quadratic",],y~x+x2)
summary(reg_quad)
```

```
Call:
lm(formula = y ~ x + x2, data = df[df$type == "quadratic", ])

Residuals:
    Min      1Q  Median      3Q     Max 
-4.7834 -0.8638 -0.0480  1.1312  3.9913 

Coefficients:
            Estimate Std. Error t value Pr(>|t|)    
(Intercept) 5.3065389  0.3348067  15.850   <2e-16 ***
x           0.0036098  0.0252807   0.143    0.887    
x2          0.0164635  0.0005694  28.913   <2e-16 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 1.803 on 47 degrees of freedom
Multiple R-squared:  0.9887,	Adjusted R-squared:  0.9882 
F-statistic:  2053 on 2 and 47 DF,  p-value: < 2.2e-16
```

Outline

Making Predictions

```r
xv <- seq(-20,60,.5)
yv <- predict(reg_quad,list(x=xv,x2=xv^2))
ggplot(data=df[df$type=="quadratic",]) + theme_bw() +
  geom_hline(yintercept=0) + geom_vline(xintercept=0) +
  geom_point(aes(x=x,y=y),color="blue") +
  geom_line(data=data.frame(x=xv,y=yv),aes(x=x,y=y),color="red",size
```



Outline

Confidence interval

Remember that

$$\hat{\beta} = \frac{\sum_{i=1}^{n} x_i y_i - \frac{1}{n} \sum_{i=1}^{n} x_i \sum_{j=1}^{n} y_j}{\sum_{i=1}^{n} (x_i^2) - \frac{1}{n} (\sum_{i=1}^{n} x_i)^2}$$

$$\hat{\alpha} = \bar{y} - \hat{\beta}\, \bar{x}$$

$\hat{\beta}$ and $\hat{\alpha}$ are sums of the ε_i's and it is thus possible to compute confidence intervals assuming:

- the linear model holds true
- either the errors in the regression are normally distributed
- or the number of observations is sufficiently large so that the actual distribution of the estimators can be approximated using the central limit theorem

Illustration with R

The Anscombe quartet

```
head(a,10)
```

```
   idx set  x    y
1    1   1 10 8.04
2    1   2 10 9.14
3    1   3 10 7.46
4    1   4  8 6.58
5    2   1  8 6.95
6    2   2  8 8.14
7    2   3  8 6.77
8    2   4  8 5.76
9    3   1 13 7.58
10   3   2 13 8.74
```

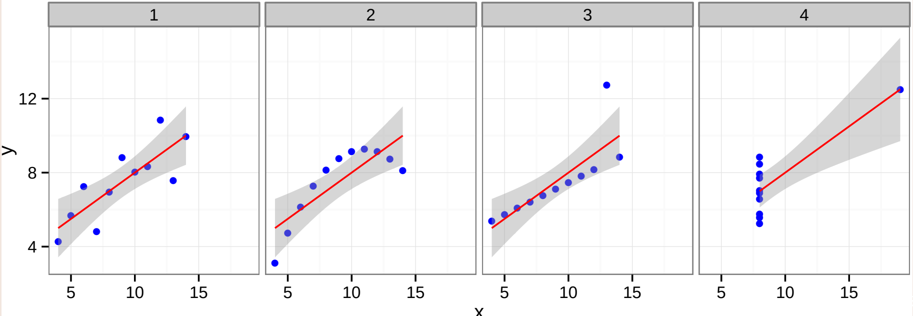
Illustration with R

The Anscombe quartet

```r
head(a,10)
```

Confidence intervals with ggplot

```r
ggplot(data=a,aes(x=x,y=y)) + theme_bw() +
  facet_wrap(~set,nrow=1) + geom_point(color="blue") +
  geom_smooth(method='lm',color="red")
```



Outline

Confidence

If we had only 1 factor with 2 levels (2^1 design), the analysis would simply amount to compute confidence intervals or more precisely to test whether $\boxed{\mu_{A=Low} = \mu_{A=High}}$ or not (t-test)

(if few observations are available we would have to make the C.I wider and use the Student distribution)

But when having more factors and/or levels, we want to test whether some of the combinations have a significantly different expected value

Number of comparisons	2	3	4	5	6
Nominal Type I error	5%	5%	5%	5%	5%
Actual overall Type I error	5%	12.2%	20.3%	28.6%	36.6%

(See 16.1.5 of *Practical Regression and Anova using R* by Julian Faraway)

Quick illustration of the difficulty of multiple testing

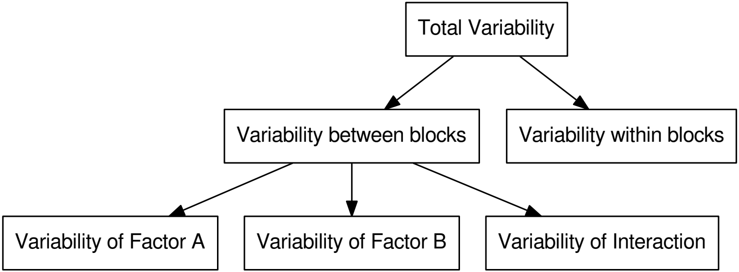
```r
se = .1; mean = 7 ;
N = 10000;
df = data.frame(
    x1=rnorm(N,mean=mean,sd=se),
    x2=rnorm(N,mean=mean,sd=se),
    x3=rnorm(N,mean=mean,sd=se))
df$eq1 = abs(df$x1-mean)<2*se
df$eq2 = abs(df$x1-df$x2)<2*se
df$eq3 = abs(df$x1-df$x2)<2*se &
         abs(df$x1-df$x3)<2*se &
         abs(df$x2-df$x3)<2*se

mean(df$eq1)
mean(df$eq2)
mean(df$eq3)
```

```
[1] 0.9538
[1] 0.8435
[1] 0.6596
```

Analysis of Variance (ANOVA)

ANOVA (ANalysis Of VAriance) enable to discriminate real effects from noise

- Enables to prove that some parameters have little influence and can be randomized over (possibly with a more elaborate model)
- Decomposes variance:



- Assumes identical standard deviation for the populations (homoscedastic)
- Multiple tests at once (assuming normality): $\boxed{\mu_{A=Low,*} - \mu_{A=High,*} = 0}$, $\boxed{\mu_{B=Low,*} - \mu_{B=High,*} = 0}$, ...

ANOVA and F-statistic

The ANOVA produces an F-statistic, the ratio of the variance calculated among the means to the variance within the samples.

- If the group means are drawn from populations with the same mean values, the variance between the group means should be lower than the variance of the samples
- A higher ratio therefore implies that the samples were drawn from populations with different mean values

ANOVA and F-statistic

The ANOVA produces an F-statistic, the ratio of the variance calculated among the means to the variance within the samples.

- If the group means are drawn from populations with the same mean values, the variance between the group means should be lower than the variance of the samples
- A higher ratio therefore implies that the samples were drawn from populations with different mean values

Let's work out a simple made-up example

```
Response = 10 + 2*as.numeric(d1$A) +
    3*as.numeric(d1$B)*as.numeric(d1$C) + rnorm(nrow(d1))
d1 <- add.response(d1,Response, replace=TRUE)
```

I had to use `as.numeric` to interpret the -1 and 1 as numbers whereas they were created as factors

A simple ANOVA in R

```r
d1_aov <- aov(Response ~ (A + B + C)^2, data=d1)
summary(d1_aov) # summary will call summary.aov
```

```
              Df Sum Sq Mean Sq F value   Pr(>F)    
A              1  22.98   22.98  38.318 0.000161 ***
B              1  68.02   68.02 113.417 2.11e-06 ***
C              1  77.60   77.60 129.402 1.21e-06 ***
A:B            1   0.44    0.44   0.728 0.415721    
A:C            1   0.93    0.93   1.555 0.243804    
B:C            1  14.62   14.62  24.374 0.000806 ***
Residuals      9   5.40    0.60                     
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

So, all factors are significant and there is a significant interaction between B and C

Can't I just read my linear regression as usual?

```
summary.lm(d1_aov)
```

```
Call:
lm.default(formula = Response ~ (A + B + C)^2, data = d1)

Residuals:
     Min       1Q   Median       3Q      Max
-1.01845 -0.48073 -0.01537  0.45886  0.98771

Coefficients:
            Estimate Std. Error t value Pr(>|t|)
(Intercept)  19.5912     0.1936 101.194 4.56e-15 ***
A1            1.1984     0.1936   6.190 0.000161 ***
B1            2.0618     0.1936  10.650 2.11e-06 ***
C1            2.2023     0.1936  11.375 1.21e-06 ***
A1:B1         0.1652     0.1936   0.853 0.415721
A1:C1         0.2415     0.1936   1.247 0.243804
B1:C1         0.9558     0.1936   4.937 0.000806 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 0.7744 on 9 degrees of freedom
Multiple R-squared:  0.9716,       Adjusted R-squared:  0.9527
F-statistic:  51.3 on 6 and 9 DF,  p-value: 1.873e-06
```

Can't I just read my linear regression as usual?

```
summary.lm(d1_aov)
```

```
Call:
lm.default(formula = Response ~ (A + B + C)^2, data = d1)
```

Wait, why is the formula so different?

$$10 + 2A + 3BC$$

```
             Estimate Std. Error t value Pr(>|t|)
(Intercept)   19.5912     0.1936 101.194 4.56e-15 ***
A1             1.1984     0.1936   6.190 0.000161 ***
B1             2.0618     0.1936  10.650 2.11e-06 ***
C1             2.2023     0.1936  11.375 1.21e-06 ***
A1:B1          0.1652     0.1936   0.853 0.415721
A1:C1          0.2415     0.1936   1.247 0.243804
B1:C1          0.9558     0.1936   4.937 0.000806 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 0.7744 on 9 degrees of freedom
Multiple R-squared:  0.9716,	Adjusted R-squared:  0.9527
F-statistic:  51.3 on 6 and 9 DF,  p-value: 1.873e-06
```

Can't I just read my linear regression as usual?

```r
summary.lm(d1_aov)
```

```
Call:
lm.default(formula = Response ~ (A + B + C)^2, data = d1)

```

Wait, why is the formula so different?

$$10 + 2A + 3BC$$

```
              Estimate Std. Error t value Pr(>|t|)
(Intercept)    19.5912     0.1936 101.194 4.56e-15 ***
A1              1.1984     0.1936   6.190 0.000161 ***
B1              2.0618     0.1936  10.650 2.11e-06 ***
C1              2.2023     0.1936  11.375 1.21e-06 ***
A1:B1           0.1652     0.1936   0.853 0.415721
A1:C1           0.2415     0.1936   1.247 0.243804
B1:C1           0.9558     0.1936   4.937 0.000806 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

eedom
ared:  0.9527
-06
```

Because it treated the factors "-1" and "1" as 0 and 1...

Then how do I get the formula I expected? (1/2)

```
d1_lm <- lm(Response ~ (as.numeric(A) + as.numeric(B) +
            as.numeric(C))^2, data=d1)
summary.aov(d1_lm)
```

```
                            Df Sum Sq Mean Sq F value   Pr(>F)
as.numeric(A)                1  22.98   22.98  38.318 0.000161 ***
as.numeric(B)                1  68.02   68.02 113.417 2.11e-06 ***
as.numeric(C)                1  77.60   77.60 129.402 1.21e-06 ***
as.numeric(A):as.numeric(B)  1   0.44    0.44   0.728 0.415721
as.numeric(A):as.numeric(C)  1   0.93    0.93   1.555 0.243804
as.numeric(B):as.numeric(C)  1  14.62   14.62  24.374 0.000806 ***
Residuals                    9   5.40    0.60
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

Sweet, it's the same as the previous ANOVA

Then how do I get the formula I expected? (2/2)

```
summary(d1_lm) # summary will call summary.lm
```

```
Call:
lm.default(formula = Response ~ (as.numeric(A) + as.numeric(B) +
    as.numeric(C))^2, data = d1)

Residuals:
     Min       1Q   Median       3Q      Max
-1.01845 -0.48073 -0.01537  0.45886  0.98771

Coefficients:
                            Estimate Std. Error t value Pr(>|t|)
(Intercept)                  15.4654     3.1870   4.853 0.000905 ***
as.numeric(A)                -0.0429     1.6878  -0.025 0.980277
as.numeric(B)                -2.6022     1.6878  -1.542 0.157516
as.numeric(C)                -2.7789     1.6878  -1.647 0.134064
as.numeric(A):as.numeric(B)   0.6606     0.7744   0.853 0.415721
as.numeric(A):as.numeric(C)   0.9658     0.7744   1.247 0.243804
as.numeric(B):as.numeric(C)   3.8232     0.7744   4.937 0.000806 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 0.7744 on 9 degrees of freedom
Multiple R-squared:  0.9716,	Adjusted R-squared:  0.9527
F-statistic:  51.3 on 6 and 9 DF,  p-value: 1.873e-06
```

Then how do I get the formula I expected? (2/2)

```r
summary(d1_lm) # summary will call summary.lm
```

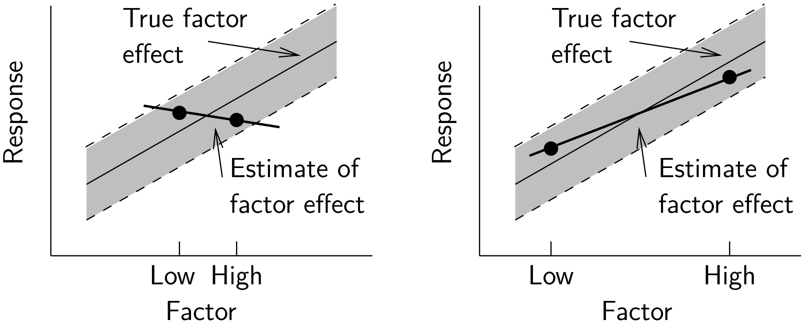
Variability is too large too obtain good estimates of the true coefficients

$$10 + 2A + 3BC$$

One should anyway use other kind of designs to estimate continuous model parameters

```
s.numeric(B) +







                             Estimate Std. Error t value Pr(>|t|)
(Intercept)                   15.4654     3.1870   4.853 0.000905 ***
as.numeric(A)                 -0.0429     1.6878  -0.025 0.980277
as.numeric(B)                 -2.6022     1.6878  -1.542 0.157516
as.numeric(C)                 -2.7789     1.6878  -1.647 0.134064
as.numeric(A):as.numeric(B)    0.6606     0.7744   0.853 0.415721
as.numeric(A):as.numeric(C)    0.9658     0.7744   1.247 0.243804
as.numeric(B):as.numeric(C)    3.8232     0.7744   4.937 0.000806 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 0.7744 on 9 degrees of freedom
Multiple R-squared:  0.9716,	Adjusted R-squared:  0.9527
F-statistic:  51.3 on 6 and 9 DF,  p-value: 1.873e-06
```

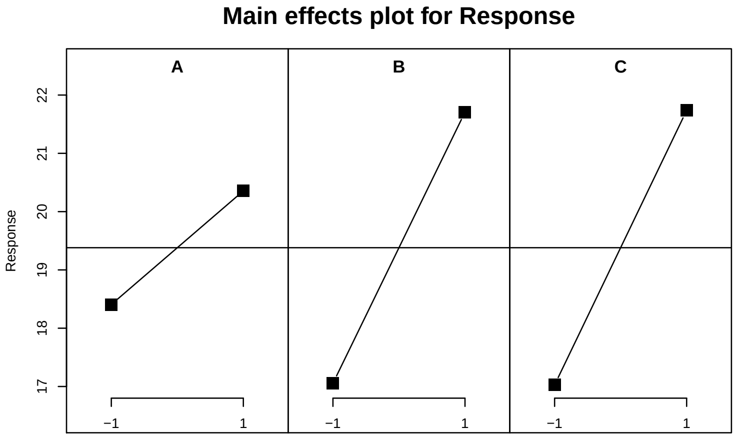
The difference between ANOVA and Linear Regression (3/3)



- The coding numbers are completely meaningless and influence the estimates of the slope
  - If your input parameters are numerical, go for *extreme* values, hoping the intermediate behavior is not too complicated and consider them as factors
- Real question: is a there significant increase when changing factors?
- Remember: you should use ANOVA for factorial designs, not LM

And graphically ?

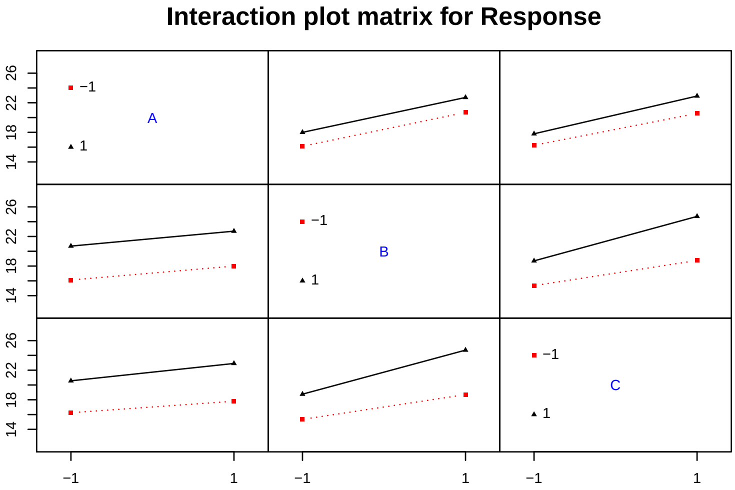
```r
MEPlot(d1, abbrev=4, select=c(1,2,3), response="Response")
```



No CI on this one but we've seen that computing CIs is not straightforward
$\rightsquigarrow$ rely on the `summary.aov`

What about interactions ?

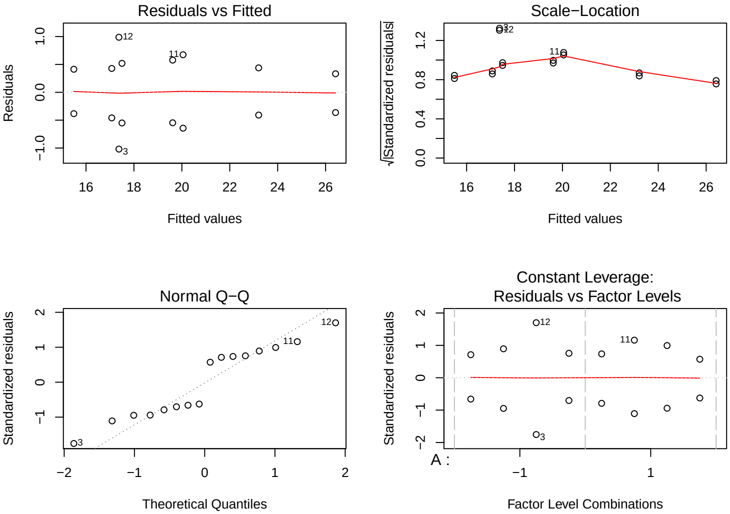
```r
IAPlot(d1, abbrev=4, show.alias=FALSE, select=c(1,2,3))
```



Again, no CI ⇝ rely on the `summary.aov`

Checking hypothesis

```r
layout(matrix(c(1,2,3,4),2,2)) # optional layout
plot(aov(Response ~ (A + B + C)^2, data=d1))
```

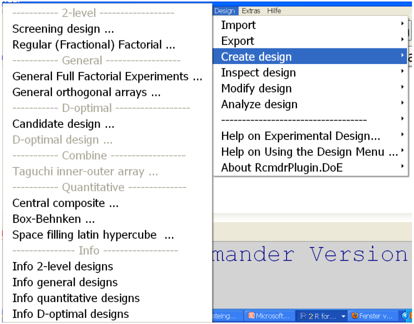


How do you expect me to ever remember all this ?

For the R commands, there is a trick: ☺

Use Rcmdr and RcmdrPlugin.DoE
(by Ulrike Grömping)

Simply `library(RcmdrPlugin.DoE)`...



You should only remember the principles and try to understand the underlying hypothesis

- ANOVA enables to discriminate real effects from noise in factorial experiments. *It relies on homoscedasticity and normality (or requires large number of samples)*
- 2-level factorial designs are a simple way to go and are more efficient than OFAT experiments
- Replicate thoroughly and randomize properly: you will not go far wrong

Outline

Linear model vs. Generalized linear Model

Linear model $Y = X.b + \varepsilon$ with $\varepsilon \sim \mathcal{N}_n(0_n, \sigma^2 I)$

$$\hat{\beta} = (X^T.X)^{-1}.X^T.Y$$

The least squares regression is the Best Linear Unbiased Estimate ($\mathrm{E}[\hat{\beta}] = b$ and $\mathrm{Var}(\hat{\beta})$ is minimal.

Unsuited when outcome Y is discrete or has unequal variance.

Generalized linear model Y is assumed to be generated from a particular *distribution* in the exponential family (normal, binomial, Poisson, gamma...)

- $\mathrm{E}[Y] = g^{-1}(X.b)$ (g is the link function)
- This allows variance to depend on the prediction ($\mathrm{Var}(Y) = g^{-1}(X.b)$)

Distribution	Link function g
Binomial ($\{0,1\}$)	$\mathrm{logit}(\mu) = \ln\left(\frac{\mu}{1-\mu}\right)$
Poisson ($\{0,1,2,\ldots\}$)	$\ln(\mu)$
Inverse Gaussian ($\mathbb{R}_+^*$)	$\frac{1}{\mu^2}$

Outline

Conclusion

1. You need a model to perform your regression
2. You need to check whether the underlying hypothesis of this model are reasonable or not

This model will allow you to:

1. Assess and quantify the effect of parameters on the response
   - Parameters are estimated as a whole, using **all** the measurements
2. Extrapolate within the range of parameters you tried
3. Detect outstanding points (those with a high residual and/or with a high lever)

This model will guide on how to design your experiments:

- e.g., the linear model assumes some uniformity of interest over the parameter space range
- if your system is heteroscedastic, you should perform more measurements for parameters that lead to higher variance