Simple linear regression and correlation

Brief history

- Correlation (Auguste Bravais 1846)
- Sir Francis Galton developed the procedure of regression and correlation during 1875-1885
- Karl Pearson – correlation coefficient in 1895

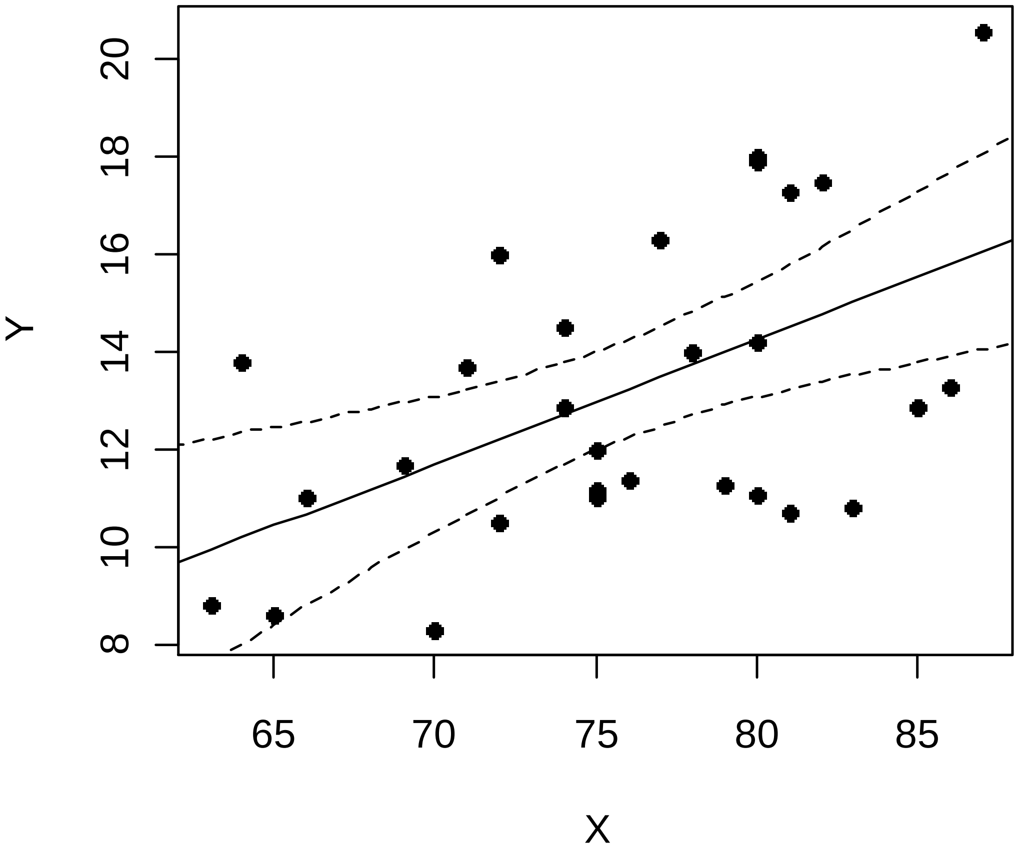
Regression vs. correlation

Two continuous variables
(simple linear regression/correlation)

- Regression
  Y-X
  - Dependent - independent
  - Effect - cause
  - Predicted - predictor
  - Response - explanatory variable / carrier
  - Output - input

- Correlation

Simple linear regression



$$Y = \alpha + \beta X + \varepsilon$$

Example

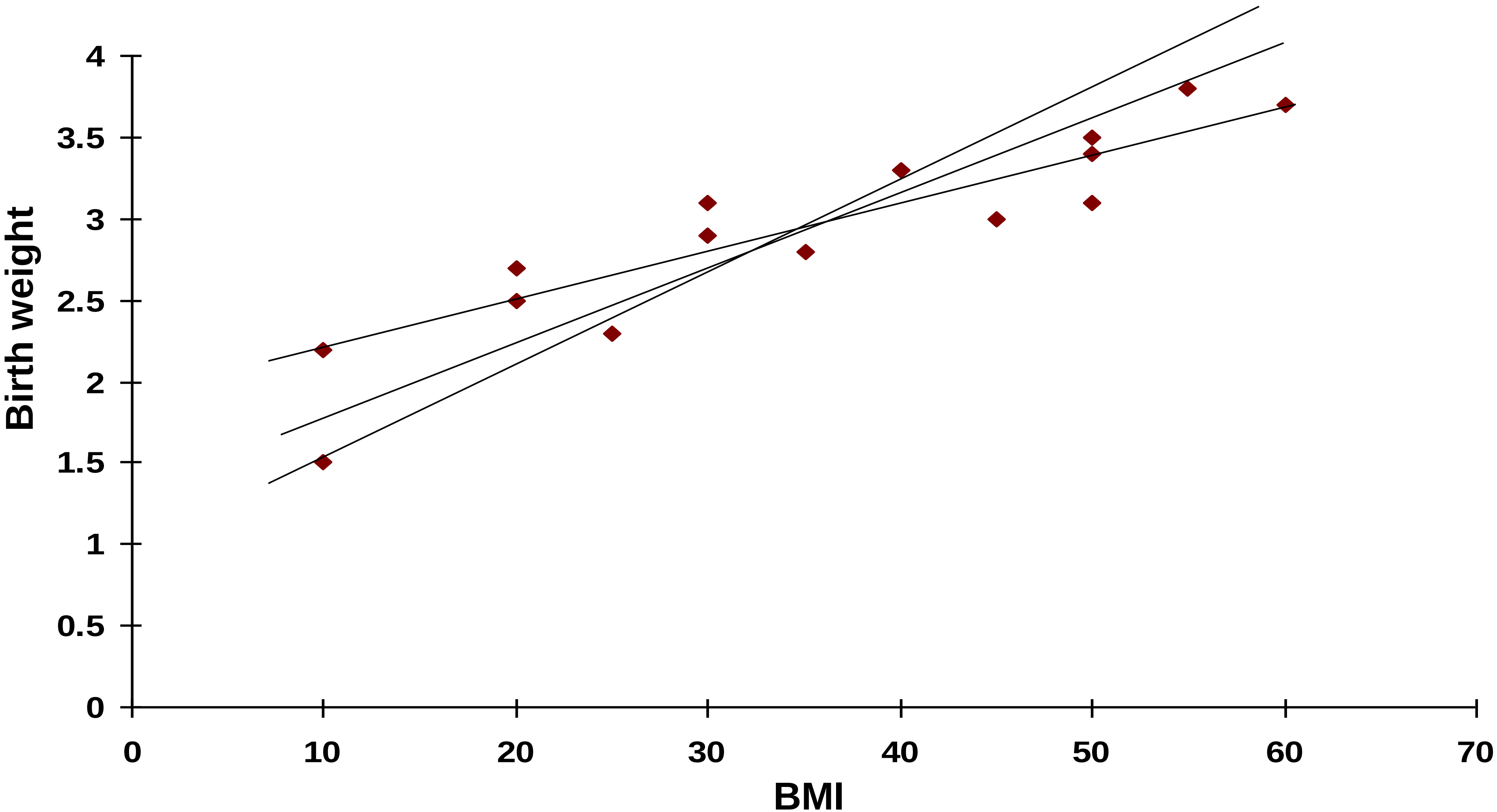
- A linear relationship between body mass index (BMI in Kg/m^2) of pregnant mothers and the birth-weight (BW in Kg) of their newborn.

- The following data set provide information on 15 pregnant mothers.

Data

Birth-weight (Kg)	BMI (Kg/m^2)
2.7	20
2.9	30
3.4	50
3.0	45
2.2	10
3.1	30
3.3	40
2.3	25
3.5	50
2.5	20
1.5	10
3.8	55
3.7	60
3.1	50
2.8	35

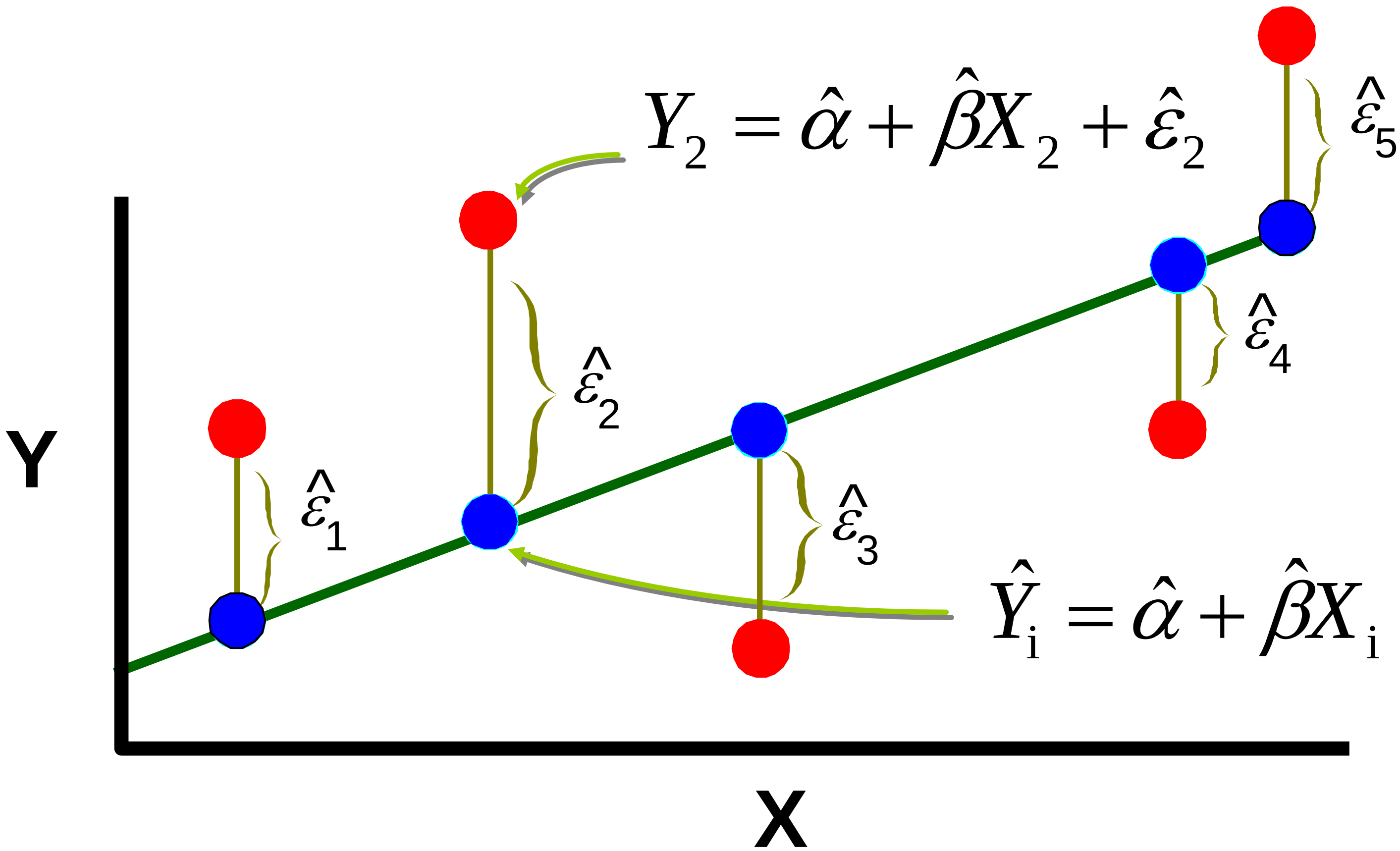
Scatter plot of BMI and birth weight



Determining the regression line

- Although we could fit a line based on visual check, but it is a subjective approach and therefore unsatisfactory.
- An objective way of determining the position of a straight line is to use the method of **least squares**.
- Using this method, we choose a line such that the sum of squares of vertical distances of all points from the line is minimized.

Least square



LS minimizes $\sum_{i=1}^{n} \hat{\varepsilon}_i^2 = \hat{\varepsilon}_1^2 + \hat{\varepsilon}_2^2 + \hat{\varepsilon}_3^2 + \hat{\varepsilon}_4^2 + \hat{\varepsilon}_5^2$

Regression line

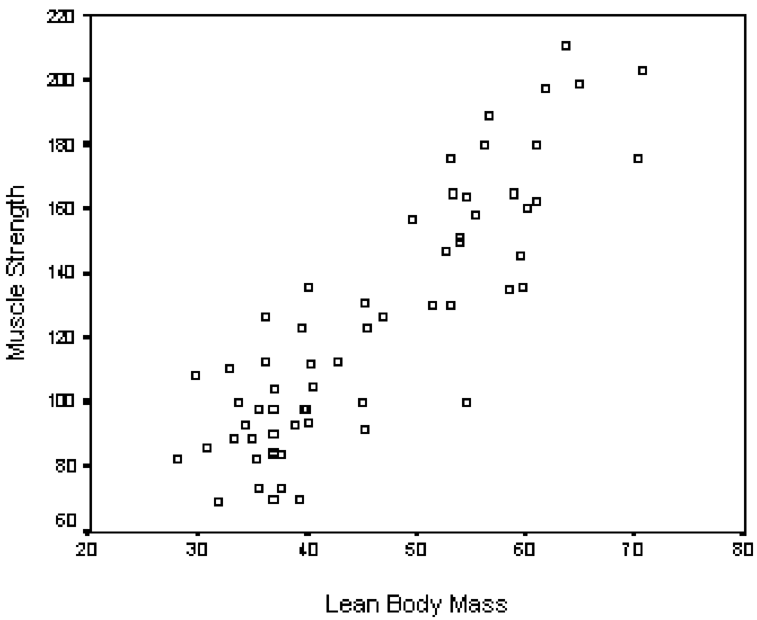
- These vertical distances, i.e., the distance between y values and their corresponding estimated values on the line are called **residuals**.

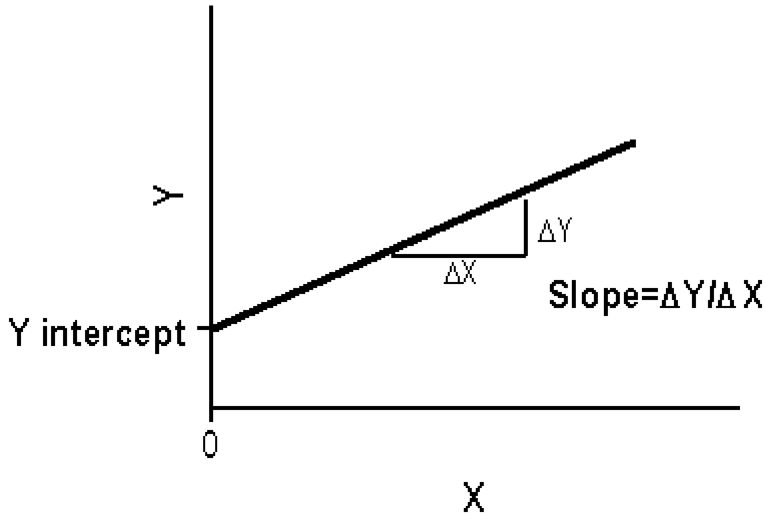
- The line which fits the best is called the regression line or, sometimes, the least-squares line.

- The line always passes through the point (X_{mean}, Y_{mean}).

Regression coefficient (slope)

$$Y = \alpha + \beta X + \varepsilon$$





α

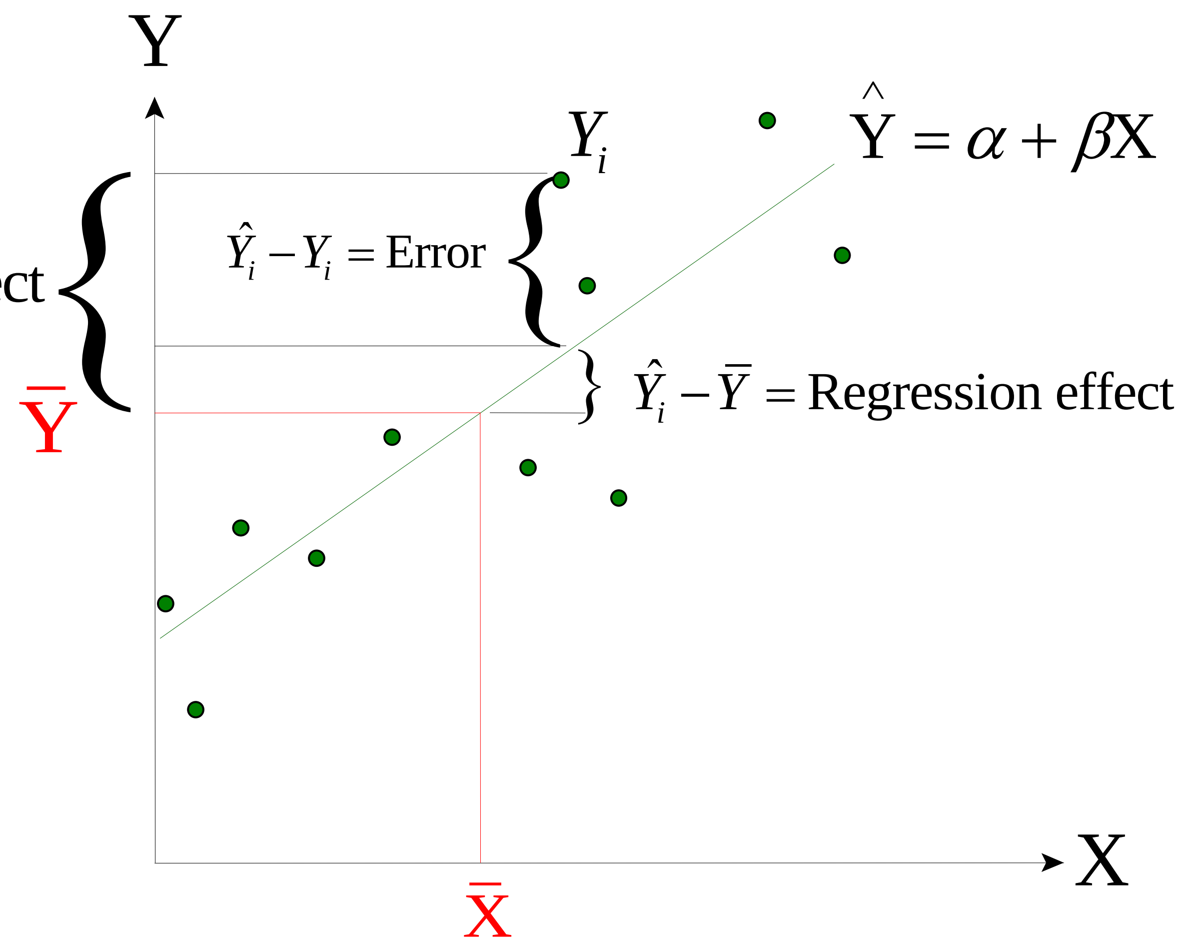
$$\beta = \frac{\sum xy}{\sum x^2}$$

$$\chi = X - \bar{X}$$

$$y = Y - \bar{Y}$$

Basic computations in regression

Decomposition of effects



$$\chi = X - \bar{X}$$

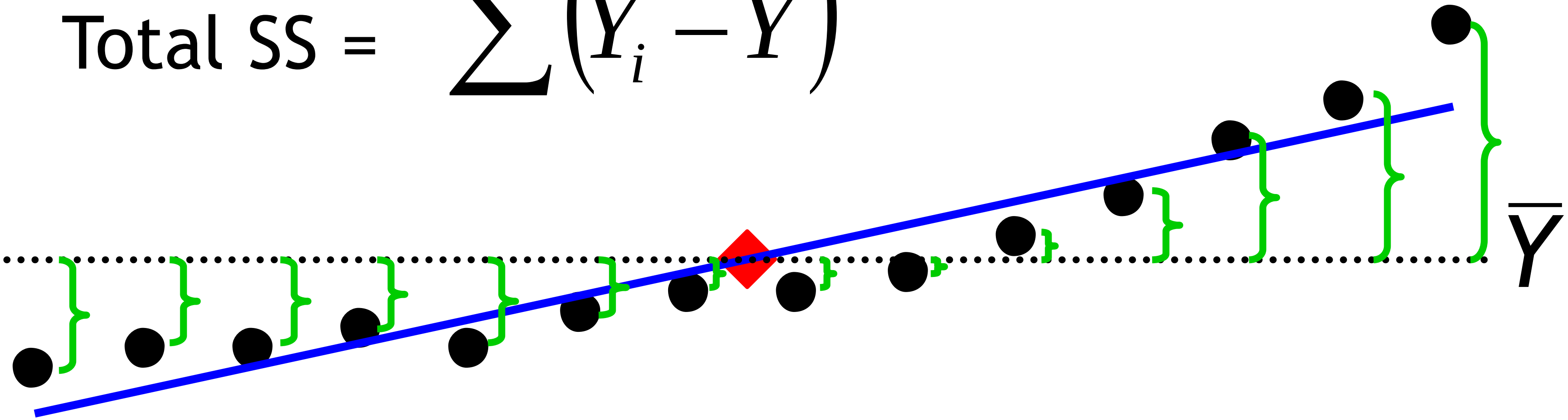
$$y = Y - \bar{Y}$$

$$\hat{Y} = \alpha + \beta X$$

$$d_{Y \cdot X} = Y - \hat{Y}$$

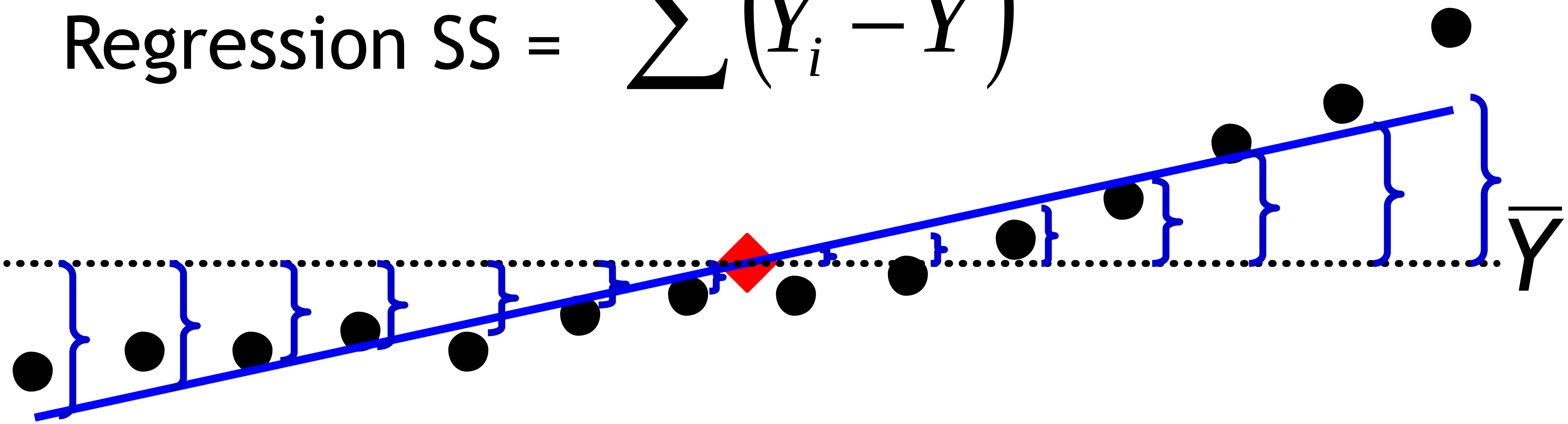
The total variation in Y (SSY)

Total SS = $\sum \left(Y_i - \overline{Y} \right)^2$



The variation in Y accounted for by the regression (SSR)

$$\text{Regression SS} = \sum\left(\hat{Y}_i - \overline{Y}\right)^2$$

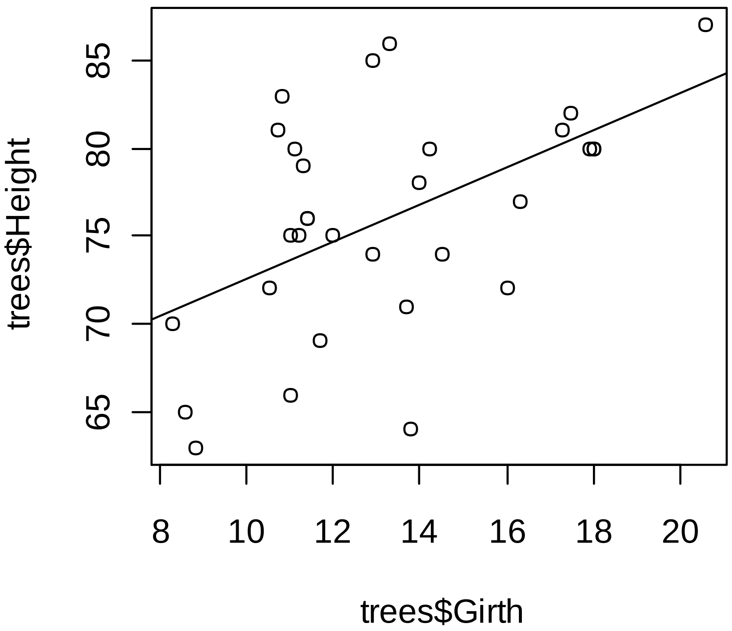


The famous five sums

trees

	Girth	Height	Volume
1	8.3	70	10.3
2	8.6	65	10.3
3	8.8	63	10.2
4	10.5	72	16.4
5	10.7	81	18.8
6	10.8	83	19.7
7	11.0	66	15.6
8	11.0	75	18.2
9	11.1	80	22.6
10	11.2	75	19.9
11	11.3	79	24.2
12	11.4	76	21.0
13	11.4	76	21.4
14	11.7	69	21.3
15	12.0	75	19.1
16	12.9	74	22.2
17	12.9	85	33.8
18	13.3	86	27.4
19	13.7	71	25.7
20	13.8	64	24.9
21	14.0	78	34.5
22	14.2	80	31.7
23	14.5	74	36.3
24	16.0	72	38.3
25	16.3	77	42.6
26	17.3	81	55.4
27	17.5	82	55.7
28	17.9	80	58.3
29	18.0	80	51.5
30	18.0	80	51.0
31	20.6	87	77.0

```
plot(trees$Girth, trees$Height)
abline(lm(trees$Height~trees$Girth))
X = trees$Girth
Y = trees$Height
```




```
# The famous five sums
sum(X)
sum(X^2)
sum(Y)
sum(Y^2)
sum(X*Y)
```

Matrix multiplication

```
sum(X);sum(X^2);sum(Y);sum(Y^2);sum(X*Y)
# matrix multiplication
XY <- cbind(1,X,Y)
t(XY) %*% XY
```

trees

	Girth	Height
1	8.3	70
2	8.6	65
3	8.8	63
4	10.5	72
5	10.7	81
6	10.8	83
7	11.0	66
8	11.0	75
9	11.1	80
10	11.2	75
11	11.3	79
12	11.4	76
13	11.4	76
14	11.7	69
15	12.0	75
16	12.9	74
17	12.9	85
18	13.3	86
19	13.7	71
20	13.8	64
21	14.0	78
22	14.2	80
23	14.5	74
24	16.0	72
25	16.3	77
26	17.3	81
27	17.5	82
28	17.9	80
29	18.0	80
30	18.0	80
31	20.6	87

```
> sum(X);sum(X^2);sum(Y);sum(Y^2);sum(X*Y)
[1] 410.7
[1] 5736.55
[1] 2356
[1] 180274
[1] 31524.7
> XY <- cbind(1,X,Y)
> t(XY) %*% XY
                X         Y
    31.0   410.70    2356.0
X  410.7  5736.55   31524.7
Y 2356.0 31524.70  180274.0
```

Sums of squares and sums of products

$$SSX = \sum (X_i - \overline{X})^2 \qquad SSY = \sum (Y_i - \overline{Y})^2 \qquad SSXY = \sum (Y_i - \overline{Y})(X_i - \overline{X})$$

```
# Sums of squares and sums of products
SSX  = sum((X-mean(X))^2); SSX
SSY  = sum((Y-mean(Y))^2); SSY
SSXY = sum((Y-mean(Y))*(X-mean(X))); SSXY
```

$$SSX = \sum (X_i)^2 - \frac{(\sum (X_i))^2}{n} \quad SSY = \sum (Y_i)^2 - \frac{(\sum (Y_i))^2}{n} \quad SSXY = \sum (X_i Y_i) - \frac{\sum (X_i) \sum (Y_i)}{n}$$

```
# The alternative way using the 5 sums
SSX  = sum(X^2)-sum(X)^2/length(X); SSX
SSY  = sum(Y^2)-sum(Y)^2/length(Y); SSY
SSXY = sum(X*Y)-sum(X)*sum(Y)/length(X); SSXY
```

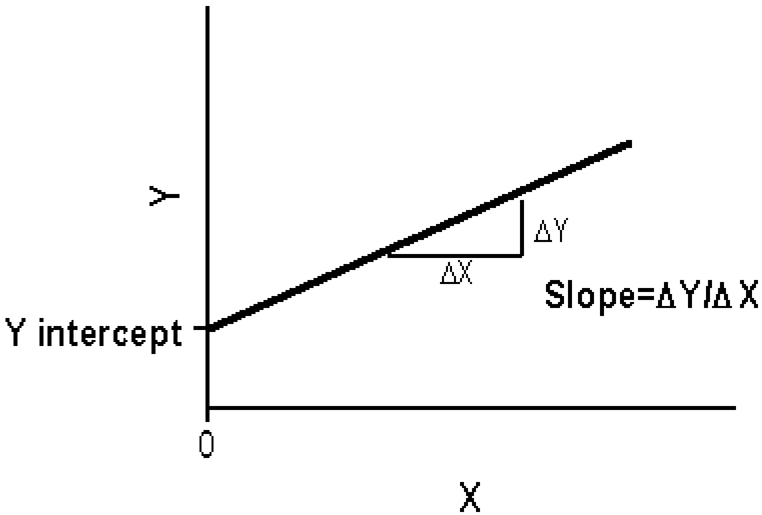
```
> SSX = sum((X-mean(X))^2); SSX
[1] 295.4374
> SSY = sum((Y-mean(Y))^2); SSY
[1] 1218
> SSXY = sum((Y-mean(Y))*(X-mean(X))); SSXY
[1] 311.5
```

Model coefficients

```
# Model (Y=a+bX) coefficients
b = SSXY/SSX; b
a = mean(Y)-b*mean(X); a
lm(Y~X)
```

```
> b = SSXY/SSX; b
[1] 1.054
> a = mean(Y)-b*mean(X); a
[1] 62.031
> lm(Y~X)
Call:
lm(formula = Y ~ X)

Coefficients:
(Intercept)           X
   62.031       1.054
```

Derivation of the Y intercept



$$y = a + bx + e$$

$$e = y - a - bx$$

$$\sum_{i=1}^{n} e_i = \sum_{i=1}^{n} y_i - \sum_{i=1}^{n} a_i - b\sum_{i=1}^{n} x_i$$

$$\text{Because by definition} \sum_{i=1}^{n} e_i = \mathbf{0}$$

$$\mathbf{0} = \sum_{i=1}^{n} y_i - \sum_{i=1}^{n} a_i - b\sum_{i=1}^{n} x_i$$

$$\sum_{i=1}^{n} a_i = \sum_{i=1}^{n} y_i - b\sum_{i=1}^{n} x_i$$

$$na = \sum_{i=1}^{n} y_i - b\sum_{i=1}^{n} x_i$$

$$a = \bar{y} - b\bar{x}$$

Derivation of the regression coefficient

$$y = a + bx + \varepsilon$$

$$\varepsilon = y - a - bx$$

$$\varepsilon_i^{\ 2} = (y_i - a - bx_i)^2$$

$$\sum \varepsilon_i^{\ 2} = \sum (y_i - a - bx_i)^2$$

$$\frac{\partial \sum \varepsilon_i^{\ 2}}{\partial b} = \frac{\partial \sum (y_i - a - bx_i)^2}{\partial b}$$

$$= -2\sum x_i (y_i - a - bx_i)$$

$$\boxed{\begin{aligned} &\bar{y} = a + b\bar{x} \\ &a = \bar{y} - b\bar{x} = \sum y_i / n - b\sum x_i / n \end{aligned}}$$

$$= -2\sum x_i (y_i - \bar{y} + b\bar{x} - bx_i)$$

$$= -2\sum x_i (y_i - \bar{y} - b(x_i - \bar{x})) = 0$$

$$b\sum x_i (x_i - \bar{x}) = \sum x_i (y_i - \bar{y})$$

$$b(\sum x_i^{\ 2} - \frac{(\sum x_i)^2}{n}) = \sum x_i y_i - \frac{(\sum x_i)(\sum y_i)}{n}$$

$$b = \frac{\sum \Delta x \Delta y}{\sum \Delta x^2} = \frac{SS_{xy}}{SS_{xx}}$$

Estimated regression line

BW ~ BMI (Birth weight ~ Body mass index)

$$\hat{y} = \hat{\alpha} + \hat{\beta}\, x = 1.775351 + 0.0330187\ x$$

$$\hat{\alpha} = 1.775351$$

$$\hat{\beta} = 0.0330187$$

Application of regression line

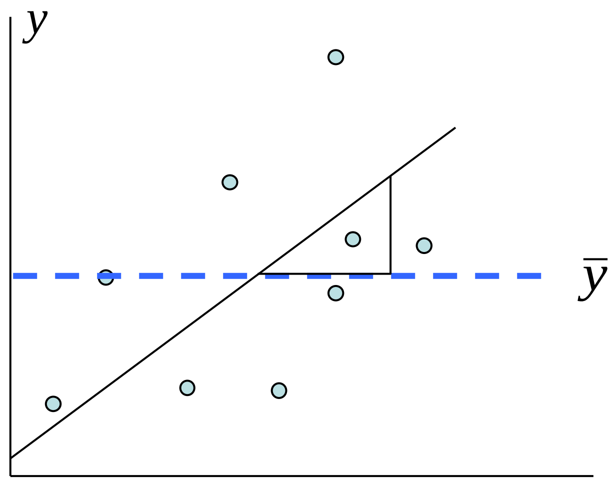
This equation allows you to estimate BW of other newborns when the BMI is given.

e.g., for a mother who has BMI=40, i.e. X = 40 we predict BW to be:

$$\hat{y} = \hat{\alpha} + \hat{\beta}\, x = 1.775351 + 0.0330187\,(40) = 3.096$$

Testing the significance of a regression

- ANOVA Testing



$$\mathrm{SSY} = \mathrm{total\,SS} = \sum (Y_i - \bar{Y})^2 = \sum y^2$$

$$\mathrm{SSR} = \mathrm{regression\,SS} = \sum (\hat{Y}_i - \bar{Y})^2$$

$$\mathrm{SSE} = \mathrm{residual\,SS} = \sum (Y_i - \hat{Y}_i)^2 = \mathrm{total\,SS} - \mathrm{regression\,SS}$$

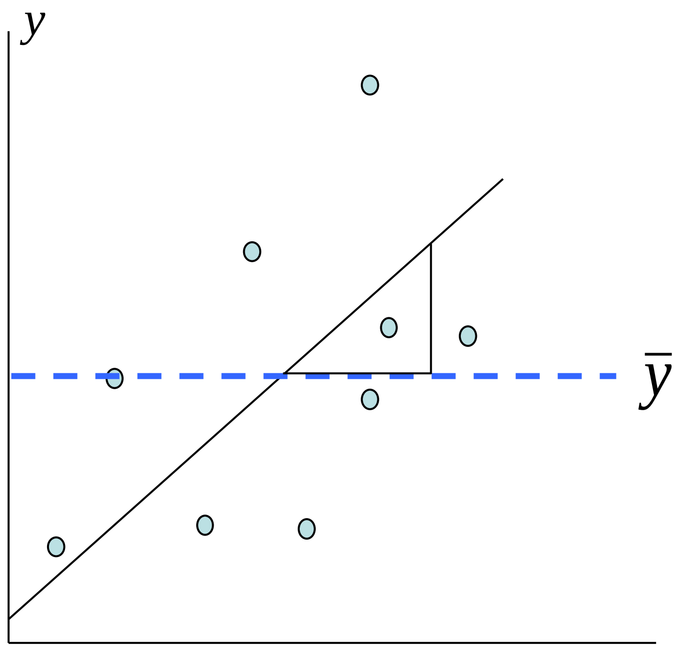
$$\mathrm{residual\,DF} = \mathrm{total\,DF} - \mathrm{regression\,DF} = n - 2$$

$$F = \frac{\mathrm{regression\,MS}}{\mathrm{residual\,MS}}$$

How many regression *df* ?

$$SSR = \sum_{i=1}^{n} (\hat{y}_i - \bar{y})^2$$

- Note that when *a* (intercept) and *b* (slope) are determined, SSR is determined.
- *a* and *b* can freely rotated in the 2d plane, hence possible *df* = 2
- Constrain: $\hat{\alpha} = \bar{y} - \hat{\beta}\,\bar{x}$
- *df*= 2-1 =1



Analysis of variance in regression

```
# Analysis of variance in regression
anova(lm(Y~X)) # data: trees
qf(0.95,1,29) # 4.18
1-pf(10.707,1,29)
```

```
> anova(lm(Y~X))
Analysis of Variance Table
Response: Y
           Df   Sum Sq   Mean Sq   F value   Pr(>F)
X           1   328.44   328.44    10.707    0.002758 **
Residuals  29   889.56    30.67
---
> qf(0.95, 1, 29)
[1] 4.182964
> 1-pf(10.707, 1, 29)
[1] 0.002757909
```

Estimation of unreliability

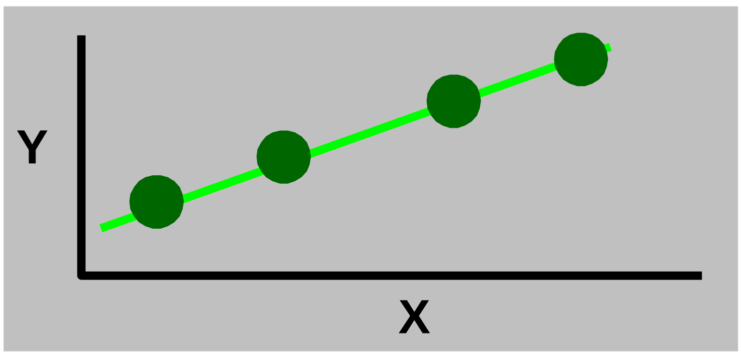
```
# Unreliability estimates for the parameters
summary(lm(Y~X))
confint(lm(Y~X))
```

```
> summary(lm(Y~X))
Residuals:
     Min       1Q    Median      3Q      Max
-12.5816  -2.7686   0.3163   2.4728   9.9456
Coefficients:
             Estimate  Std. Error  t value   Pr(>|t|)
(Intercept)   62.0313      4.3833   14.152   1.49e-14 ***
X              1.0544      0.3222    3.272   0.00276 **

Residual standard error: 5.538 on 29 degrees of freedom
Multiple R-squared: 0.2697,    Adjusted R-squared: 0.2445
F-statistic: 10.71 on 1 and 29 DF,  p-value: 0.002758

> confint(lm(Y~X))
                 2.5 %     97.5 %
(Intercept) 53.0664541  70.996174
X            0.3953483   1.713389
```

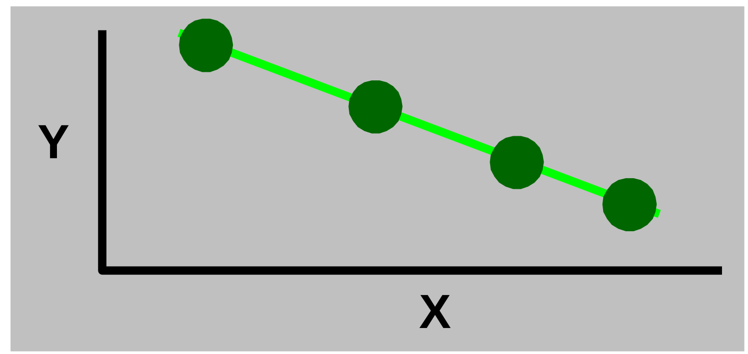
Coefficient of determination (R square)

$$r^2 = \frac{\text{regression SS}}{\text{total SS}} = \frac{SSR}{SSY}$$

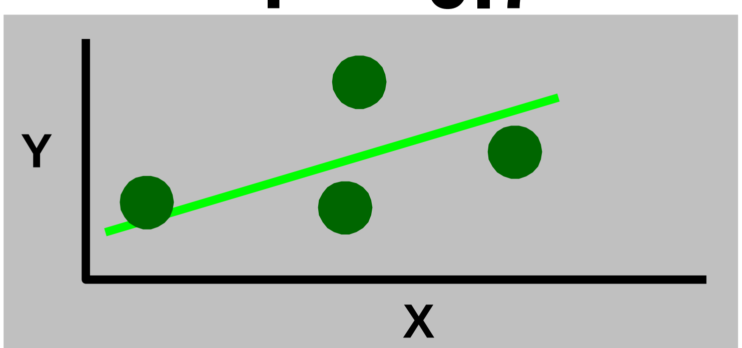
$r^2 = 1$



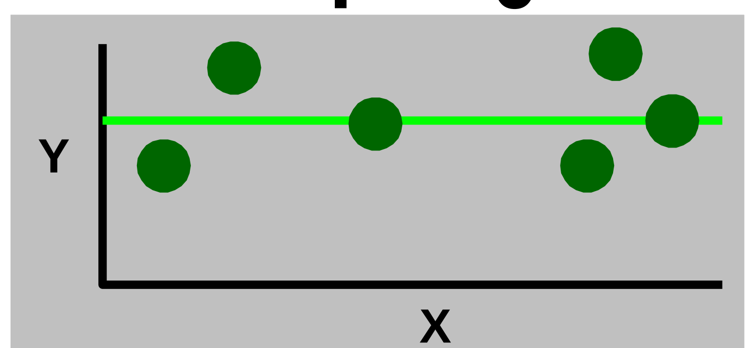
$r^2 = 1$



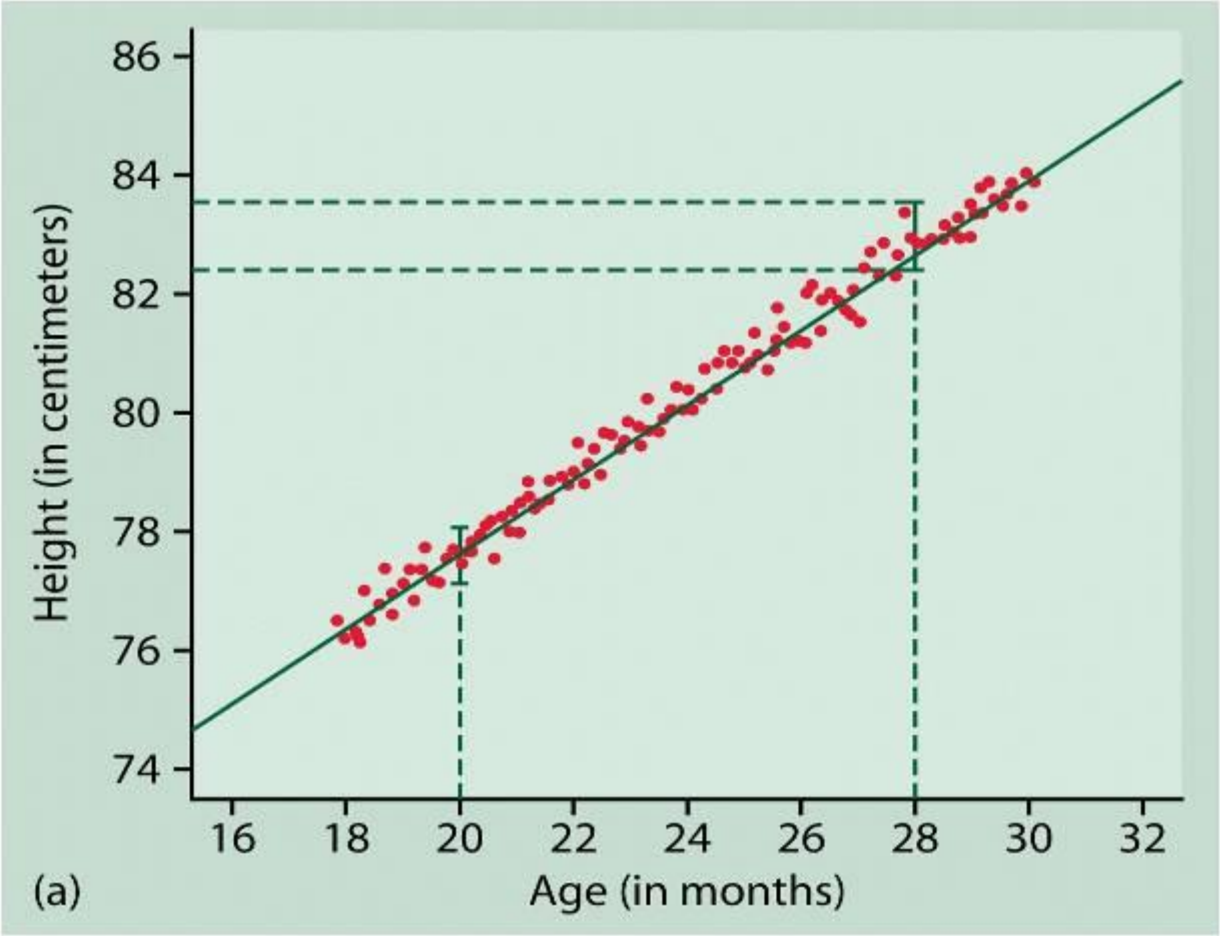
$r^2 = 0.7$



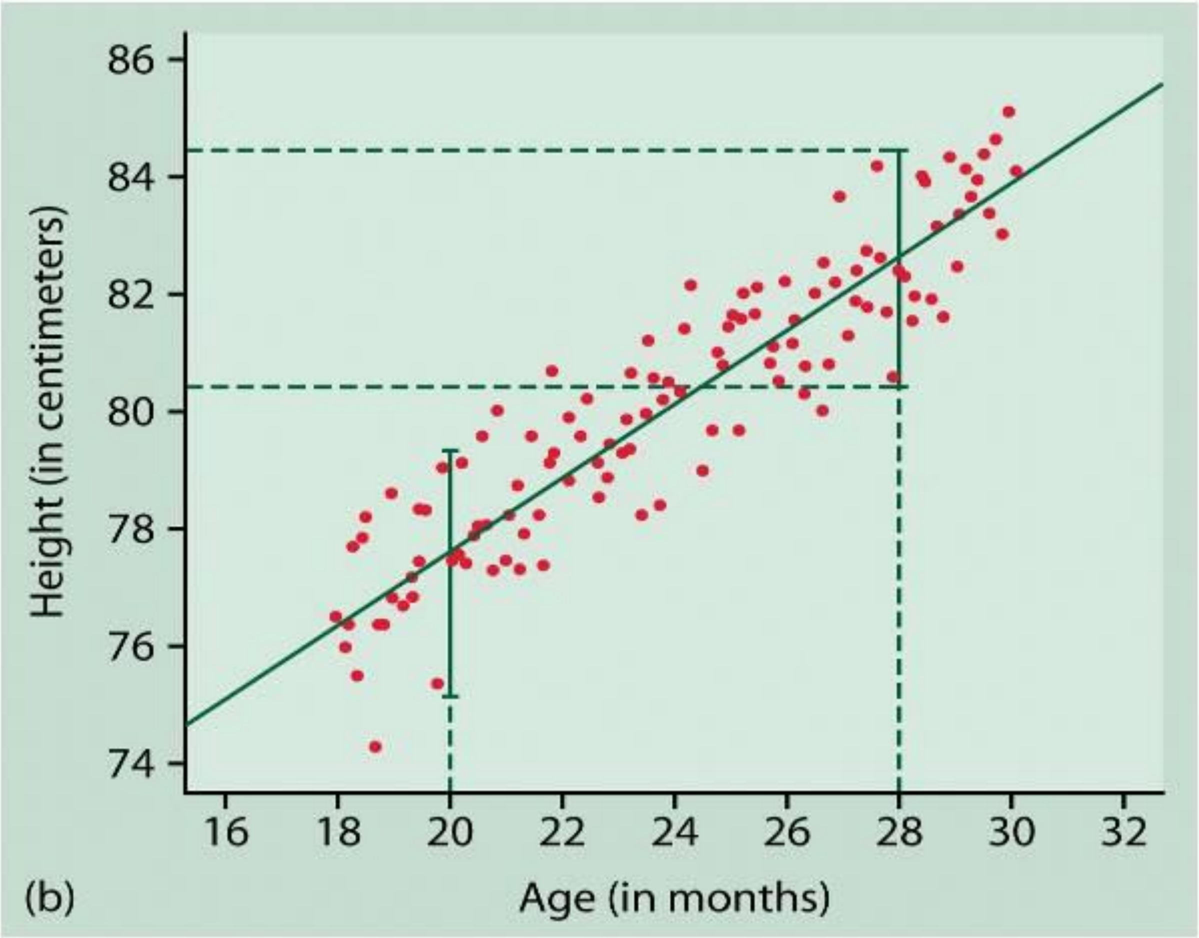
$r^2 = 0$



Age vs. Height: r^2=0.988



Age vs. Height: r^2=0.849



Degree of scatter

```
# Degree of scatter
SSY = deviance(lm(Y~1)); SSY # SSY
SSE = deviance(lm(Y~X)); SSE # SSE
rsq = (SSY-SSE)/SSY; rsq # R square
summary(lm(Y~X))[[8]]
```

```
> deviance(lm(Y~1)) # SSY
[1] 1218
> SSE = deviance(lm(Y~X)); SSE # SSE
[1] 899.5451
> rsq = (SSY-SSE)/SSY; rsq # R square
[1] 0.2696518
> summary(lm(Y~X))[[8]]
[1] 0.2696518
```

R^2 Computer Output

```
summary(lm(Volume~Girth, trees))$r.squared
```

```
[1] 0.9353199
```

```
summary(lm(Volume~Girth, trees))$adj.r.squared
```

```
[1] 0.9330895
```

R square and adjusted R square

- R square is a non-decreasing function of the number of variables in the model.
- R square therefore can not be compared of 2 models unless both have same number of variables and also the dependent variable appears in both models in the same form.
- To enable comparison of R square of different models with different numbers of variables we use Adjusted R square.

$$R^2 = \frac{\text{Explained Variation}}{\text{Total Variation}}$$

$$= \frac{\sum_{i=1}^{n}\left(Y_i - \bar{Y}\right)^2 - \sum_{i=1}^{n}\left(Y_i - \hat{Y}\right)^2}{\sum_{i=1}^{n}\left(Y_i - \bar{Y}\right)^2} = \frac{\sum_{i=1}^{n}\left(\hat{Y} - \bar{Y}\right)^2}{\sum_{i=1}^{n}\left(Y_i - \bar{Y}\right)^2}$$

adj $R^2 = 1 - (1 - R^2)\frac{n-1}{n-k}$

K is the number of coefficients (one intercept and a number of slopes)

Regression performance – R square and F statistic

- R^2 measures the proportion of explained variance
- F statistic measure the explained over unexplained

$$F = \frac{MSR}{MSE}$$

Testing the significance of the slope

t test

- H_0: $\beta = \beta_0$
- H_A: $\beta \neq \beta_0$

- $\beta_0 \neq 0$, *or* $\beta_0 = 0$

$$t = \frac{\text{(parameter estimated)} - \text{(parameter value hypothesized)}}{\text{standard error of parameter estimated}}$$

$$t = \frac{b - \beta_0}{s_b} \qquad s_b^2 = \frac{\frac{1}{n-2}\sum (y_i - \hat{y})^2}{\sum (x_i - \bar{x})^2}$$

Confident interval for the regression coefficient

$$t = \frac{b - \beta_0}{s_b} \qquad b \pm t_{\alpha(2),\,(n-2)} s_b$$

$$s_b^2 = \frac{\frac{1}{n-2}\sum (y_i - \hat{y})^2}{\sum (x_i - \bar{x})^2}$$

Standard errors of regression coefficients

```
summary(lm(Y~X))[[4]][4] # The standard error of the slope
summary(lm(Y~X))[[4]]
```

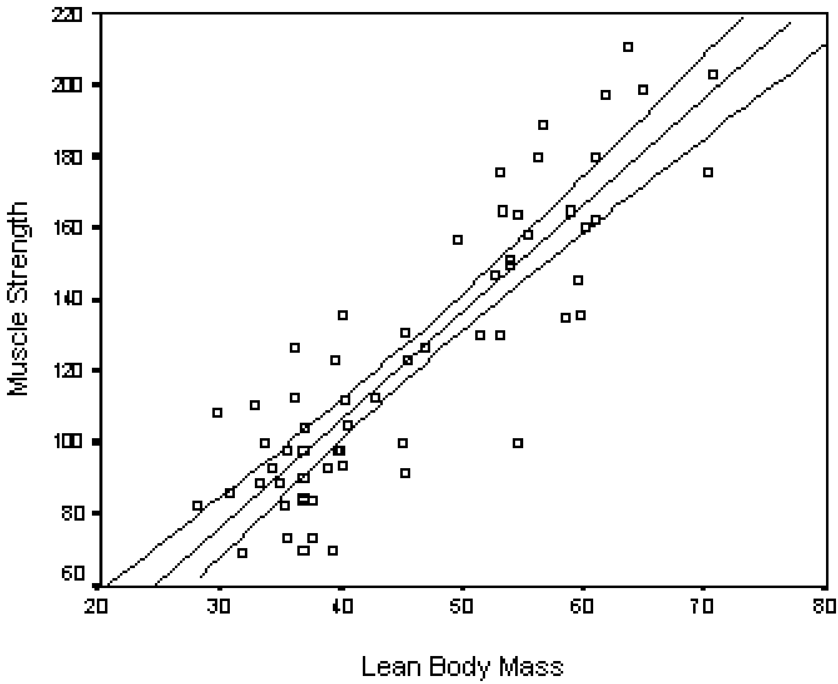
```
> summary(model)[[4]][4]
[1] 0.3222233
> summary(model)[[4]]
              Estimate   Std. Error    t value     Pr(>|t|)
(Intercept)  62.031314    4.3833023  14.151731  1.492972e-14
X             1.054369    0.3222233   3.272169  2.757815e-03
```

Confidence interval for an estimated y_j at x_j

$$E(y) \pm t_{n-2,\alpha(2)} \cdot S_{\hat{y}}$$

$$S_{\hat{y}_j} = \sqrt{\frac{\sum(y_i - \hat{y})^2}{n-2}\left[\frac{1}{n} + \frac{(x_j - \bar{x})^2}{\sum(x_i - \bar{x})^2}\right]}$$

DF = n-2



Factors affecting interval width

1. Level of Confidence (1 - α)

   Width decreases as confidence (1 - α) decreases

2. Sample Size

   Width decreases as sample size increases

3. Error variance (*SSE*)

   Width increases as variation increases

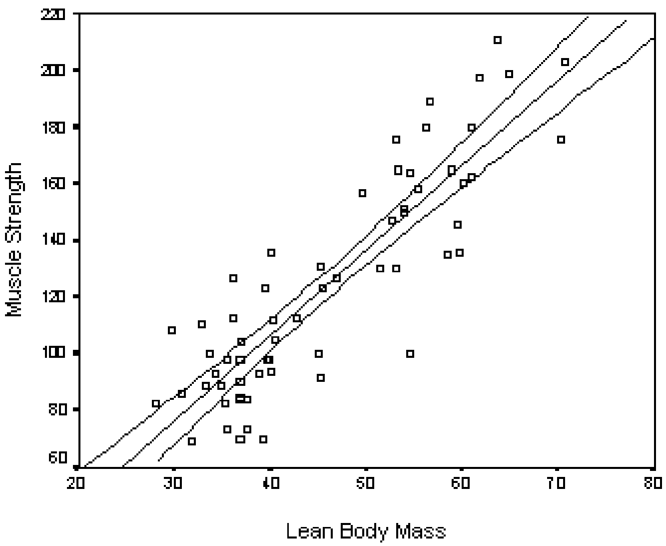
4. Distance of X from Mean X

   Width decreases as distance decreases

5. Data Dispersion (*max*(*x*)-*min*(*x*))

   Width decreases as dispersion increases

$$S_{\hat{y}_j}=\sqrt{\frac{\sum(y_i-\hat{y})^2}{n-2}\left[\frac{1}{n}+\frac{(x_j-\bar{x})^2}{\sum(x_i-\bar{x})^2}\right]}$$



Interval estimate computer output

Obs	Dep Var SALES	Pred Value	Std Err Predict	Low95% Mean	Upp95% Mean	Low95% Predict	Upp95% Predict
1	1.000	0.600	0.469	-0.892	2.092	-1.837	3.037
2	1.000	1.300	0.332	0.244	2.355	-0.897	3.497
3	2.000	2.000	0.271	1.138	2.861	-0.111	4.111
4	2.000	2.700	0.332	1.644	3.755	0.502	4.897
5	4.000	3.400	0.469	1.907	4.892	0.962	5.837

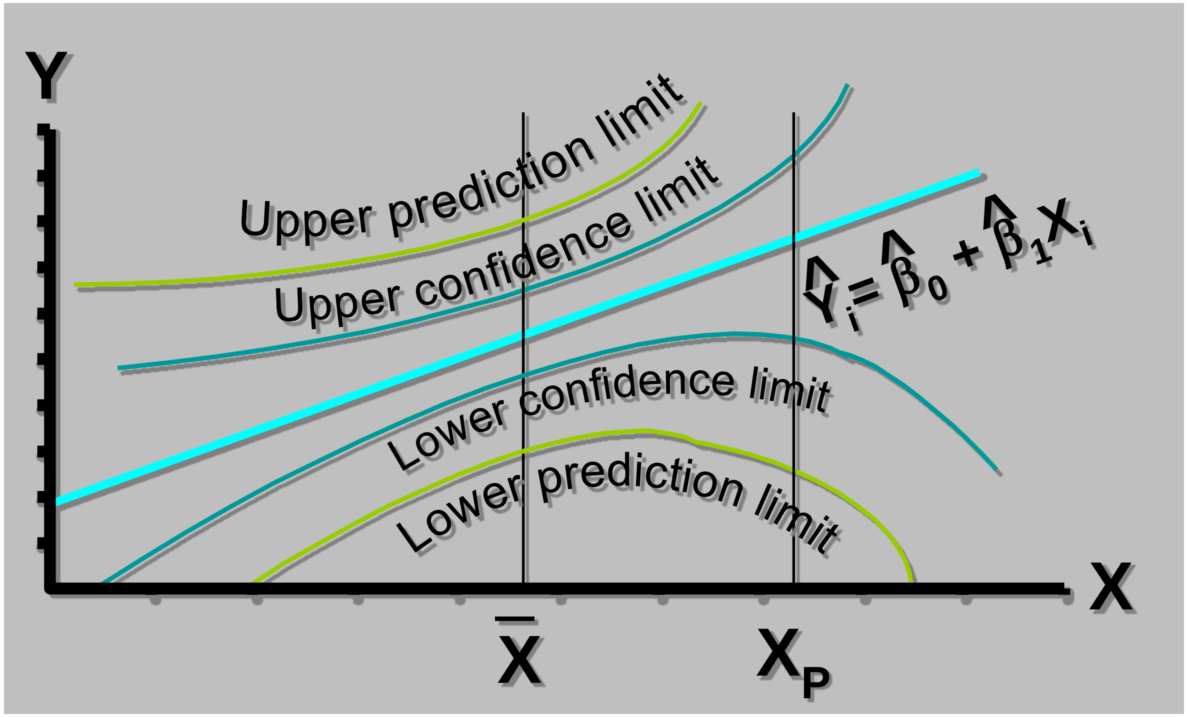
Predicted *Y* when *X* = 4

$S_{\hat{Y}}$

Confidence Interval

Prediction Interval

Hyperbolic interval bands



Prediction: calculate the "true" value of the dependent variable at values of the independent variables that we have not measured.

$$\text{Confidence interval} = \hat{y}_j \pm t_{n-2,\alpha(2)} \times S/\sqrt{n} = \hat{y}_j \pm t_{n-2,\alpha(2)} \times S_{\hat{y}_j}$$

$$\text{Prediction interval} = \hat{y}_j \pm t_{n-2,\alpha(2)} \times S \times \sqrt{1 + 1/n}$$

$$S_{\hat{y}_j} = \sqrt{\frac{\sum(y_i - \hat{y})^2}{n-2}\left[\frac{1}{n} + \frac{(x_j - \bar{x})^2}{\sum(x_i - \bar{x})^2}\right]}$$

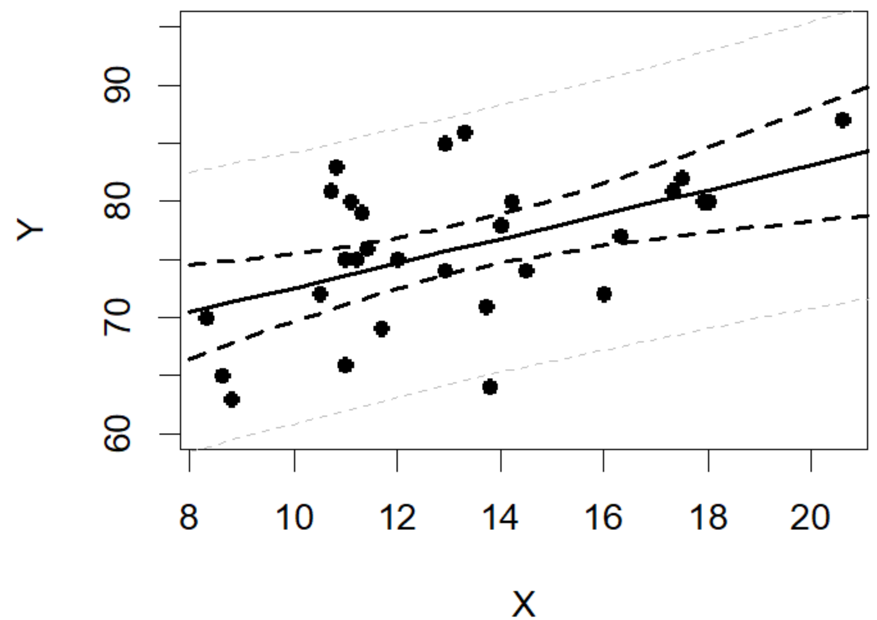
Prediction using the fitted model

```
# Prediction using the fitted model
model <- lm(Y~X)
predict(model, list(X = c(14,15,16)))
```

```
> model<-lm(Y~X)
> predict(model,list(X=c(14,15,16)))
       1        2        3
76.79248 77.84685 78.90121
```

Plot estimation CI

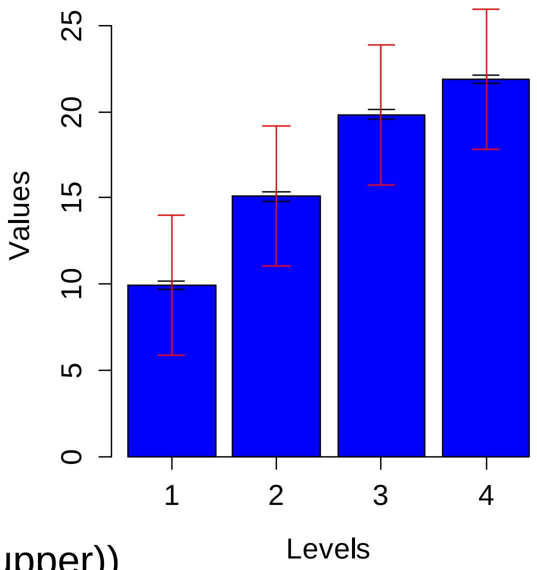
```
ci.lines<-function(model){
  xm <- mean(model[[12]][,2])
   n <- length(model[[12]][[2]])
  ssx<- sum(model[[12]][2]^2)- sum(model[[12]][2])^2/n
  s.t<- qt(0.975,(n-2))
  xv <- seq(min(model[[12]][2]),max(model[[12]][2]),
        (max(model[[12]][2])-min(model[[12]][2]))/100)
  yv <- coef(model)[1]+coef(model)[2]*xv
  se <- sqrt(summary(model)[[6]]^2*(1/n+(xv-xm)^2/ssx))
  ci <- s.t * se
  uyv<- yv + ci
  lyv<- yv - ci
  lines(xv, uyv, lty=2)
  lines(xv, lyv, lty=2)
}
plot(X, Y, pch = 16)
abline(model)
ci.lines(model)
```




```
# Another method
X = trees$Girth; Y = trees$Height
model <- lm(Y~X)
plot(X, Y, pch = 16, ylim=c(60,95)))
xv <- seq(8,22,1)
y.c <- predict(model,list(X=xv),int="c")  # "c": 95% CI
y.p <- predict(model,list(X=xv),int="p")  # "p": prediction
matlines(xv, y.c, lty=c(1,2,2), lwd=2, col="black")
matlines(xv, y.p, lty=c(1,2,2), lwd=1, col=c("black","grey","grey"))
```

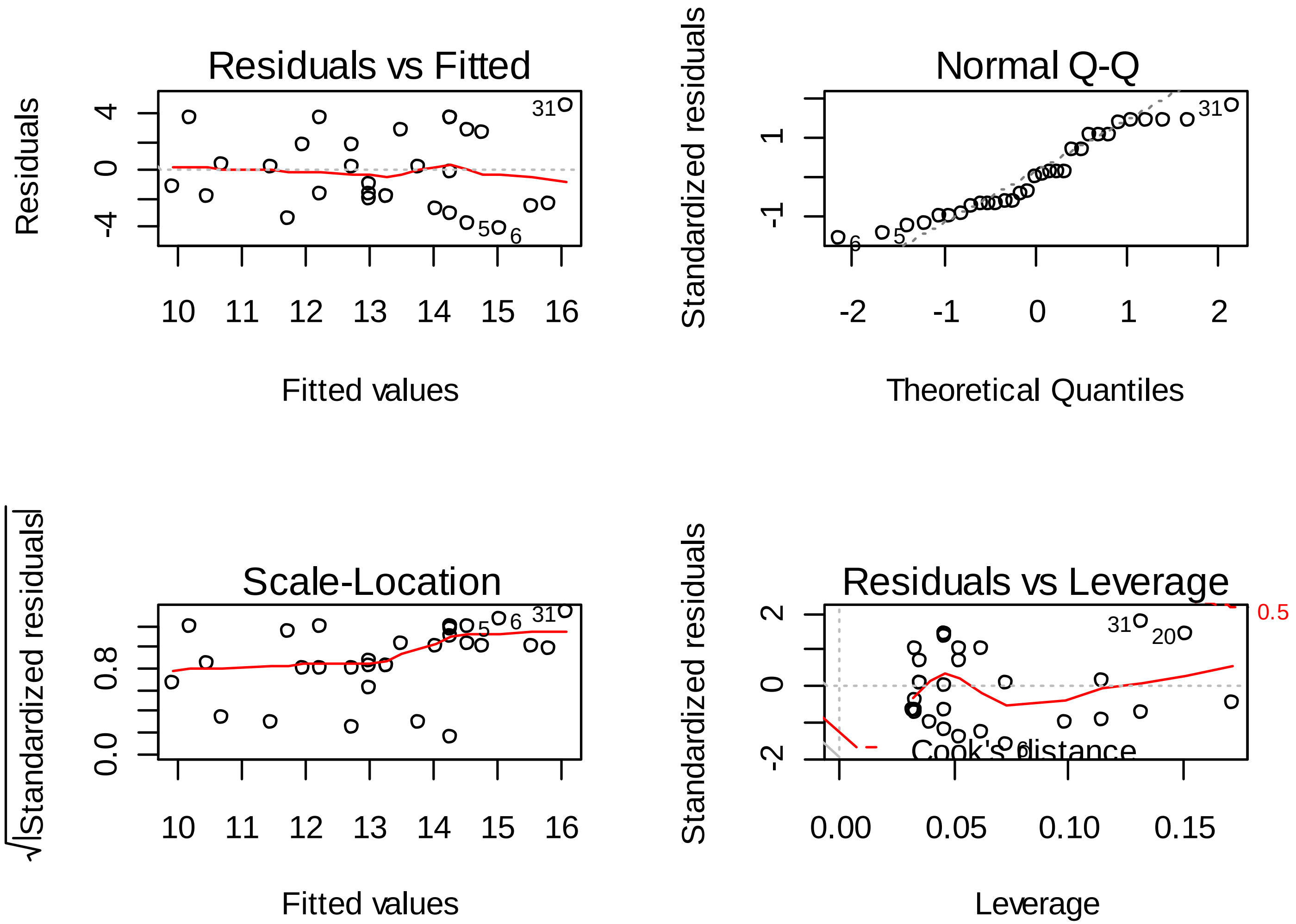
Error bars (for categorical levels)

```
x1 <- rep( 0:1, each=500 ); x2 <- rep( 0:1, each=250, length=1000 )
y <- 10 + 5*x1 + 10*x2 - 3*x1*x2 + rnorm(1000,0,2) #values
fit1 <- lm( y ~ x1*x2 )
newdat <- expand.grid( x1=0:1, x2=0:1 )
pred.lm.ci <- predict(fit1, newdat, interval='confidence')
pred.lm.pi <- predict(fit1, newdat, interval='prediction')
pred.lm.ci; pred.lm.pi

# function for plotting error bars from http://monkeysuncle.stanford.edu/?p=485
error.bar <- function(x, y, upper, lower=upper, length=0.1,...){
  if(length(x) != length(y) | length(y) !=length(lower) | length(lower) != length(upper))
  stop("vectors must be same length")
  arrows(x,y+upper, x, y-lower, angle=90, code=3, length=length, ...)
}

barx <- barplot(pred.lm.ci[,1], names.arg=1:4, col="blue", axis.lty=1, ylim=c(0,28),
xlab="Levels", ylab="Values")
# Error bar for confidence interval
error.bar(barx, pred.lm.ci[,1], pred.lm.ci[,2]-pred.lm.ci[,1],pred.lm.ci[,1]-pred.lm.ci[,3])
# Error bar for prediction interval
error.bar(barx, pred.lm.pi[,1], pred.lm.pi[,2]-pred.lm.pi[,1],pred.lm.pi[,1]-pred.lm.pi[,3],col='red')
```



Model checking

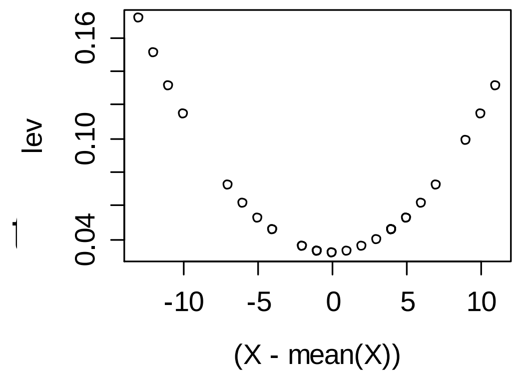
```
# Model checking
par(mfrow=c(2,2));    plot(model)
```



Influence

- An observation's influence is a function of two factors: (1) leverage, (2) distance.

Leverage



- The leverage of an observation is based on how much t differs from the mean of the predictor variable.

- The greater an observation's leverage, the more potential it has to be an influential observation.

  – For example, an observation with a value equal to the mean on the predictor variable has no influence on the slope of the regression line regardless of its value on the criterion variable.
  – On the other hand, an observation that is extreme on the predictor variable has the potential to affect the slope greatly.

In linear regression model, the leverage score (self-sensitivity or self-influence) for the data unit *i* is defined as:

$$h_{ii} = \frac{\partial \hat{y}_i}{\partial y_i}$$

```
leverage <- hat(model.matrix(model))
```

Calculation of Leverage (h)
The first step is to standardize the predictor variable (mean=0, SD=1). Then, the leverage (h) is computed by squaring the observation's value on the standardized predictor variable, adding 1, and dividing by the number of observations. $(X_i^2+1)/n$

Distance

- The distance of an observation is based on the error of prediction for the observation: The greater the error of prediction, the greater the distance.

Cook's distance

cooks.distance(model)

- Estimate the **influence** of a data point when performing least squares regression analysis. It is named after the American statistician R. Dennis Cook, who introduced the concept in 1977.

- Cook's distance measures the effect of deleting a given observation. Data points with large residuals (outliers) and/or high leverage may distort the outcome and accuracy of a regression. Points with a large Cook's distance (D>1) are considered to merit closer examination in the analysis.

$$D_i = \frac{\sum_{j=1}^{n} (\hat{Y}_j - \hat{Y}_{j(i)})^2}{p \times MSE}$$

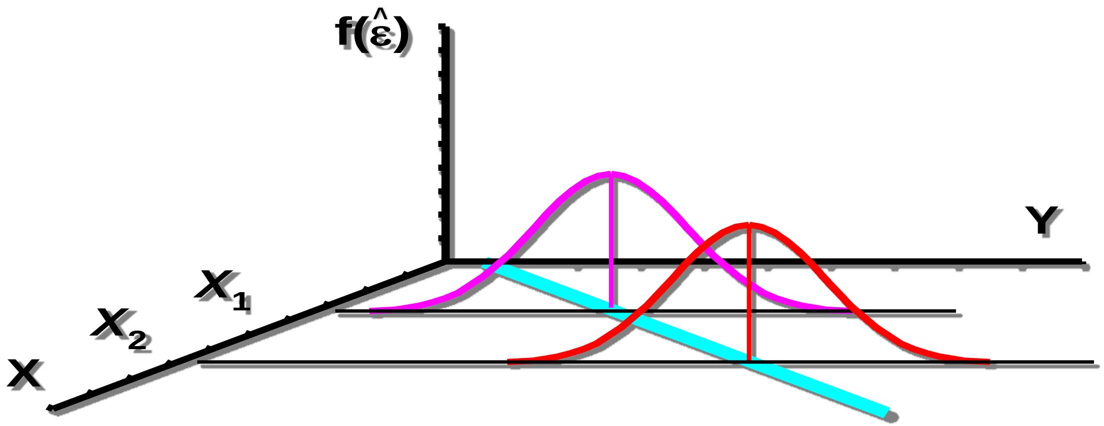
Y hat$_{j(i)}$ is the prediction for observation *j* from a refitted regression model in which observation *i* has been omitted. *P* is the number of fitted parameters in the model.

Model update

```r
# Model update (remove one outlier)
model2 <- update(model,subset=(X != 15))
summary(model2)

# Slope
coef(model2)[2] # 1.054369
model2$coefficients[2] # 1.054369
```

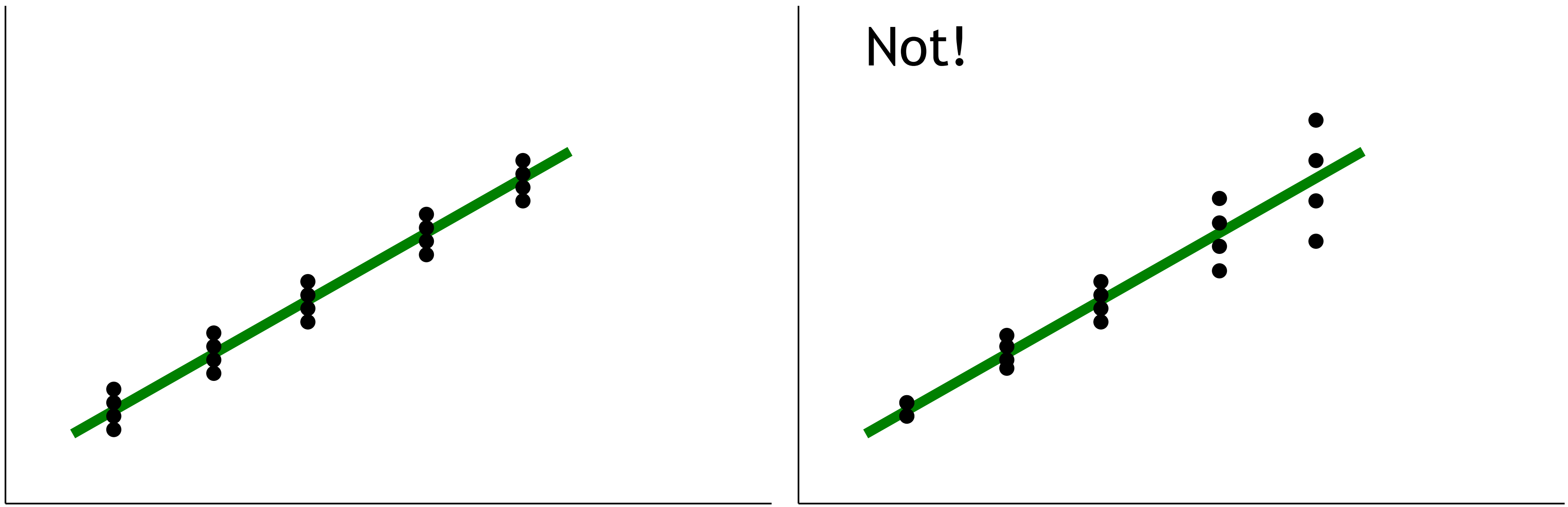
Assumptions of regression analysis

- Normal distribution of Y for value of X
- Homogeneity of variance
- The actual relationship is linear
- Values of Y are independent to each other
- X has no error
    - Zar p332

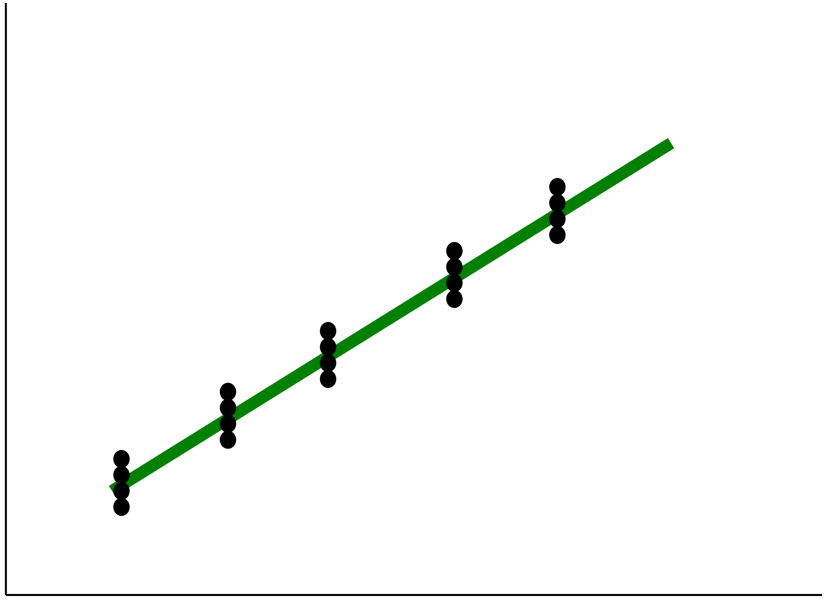


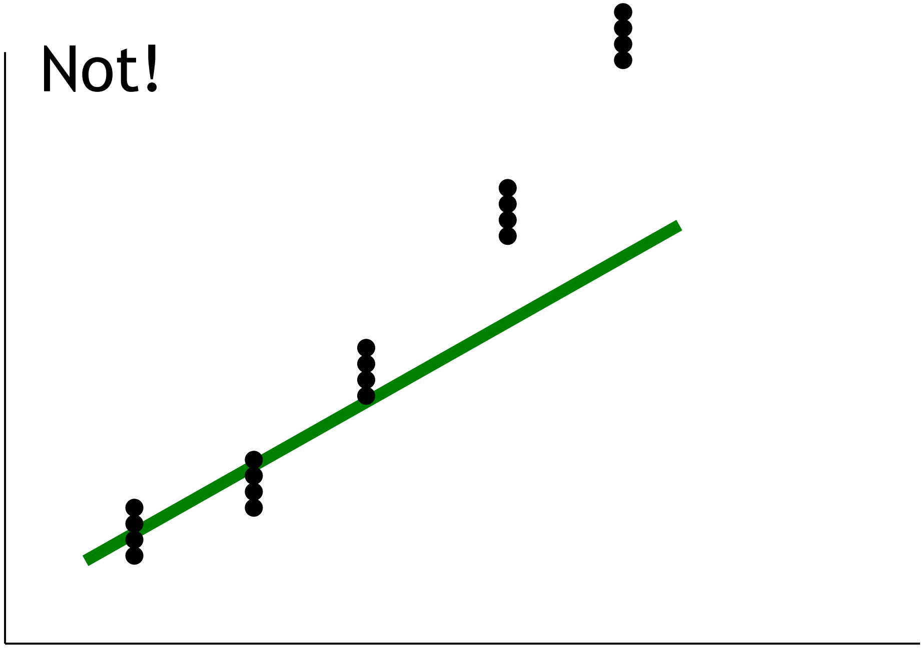
Equality of variances of Ys across values of X

(i.e., values of residuals are not related to values of X)

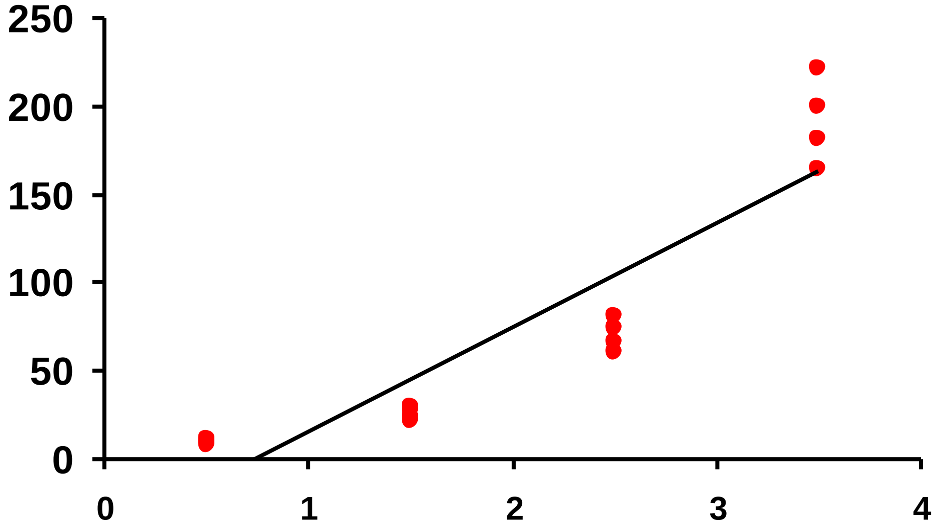


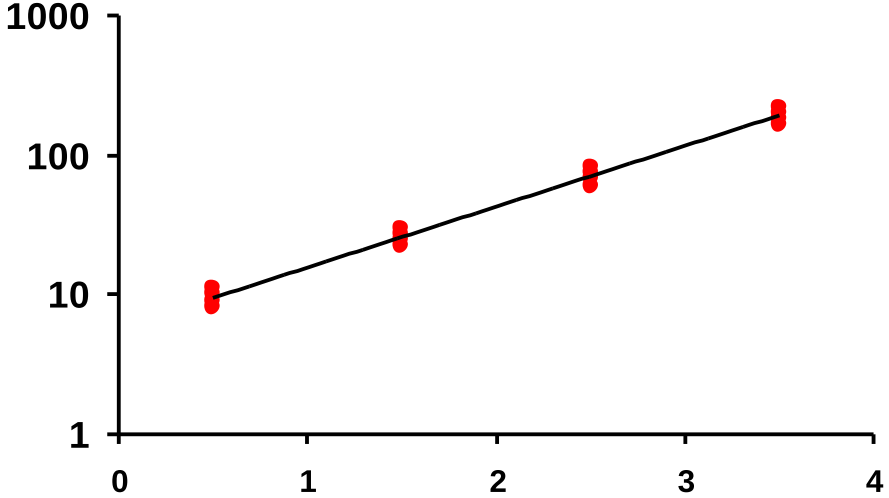
Actual relationship is linear





Log-transforming the data *may* improve the situation





Regression with replication

Do not use mean

Age (yr) X	*Systolic blood pressure* (mm Hg) Y	n_i
30	108, 110, 106	3
40	125, 120, 118, 119	4
50	132, 137, 134	3
60	148, 151, 146, 147, 144	5
70	162, 156, 164, 158, 159	5

$$N = 20$$

$$\sum\sum X_{ij} = 1050 \qquad \sum\sum Y_{ij} = 2744$$

$$\sum\sum X_{ij}^2 = 59{,}100 \qquad \sum\sum Y_{ij}^2 = 383{,}346 \qquad \sum\sum X_{ij}Y_{ij} = 149{,}240$$

$$\sum x^2 = 3975.00 \qquad \sum y^2 = 6869.20 \qquad \sum xy = 5180.00$$

$$\bar{X} = 52.5 \qquad \bar{Y} = 137.2$$

$$b = \frac{\sum xy}{\sum x^2} = \frac{5180.00}{3975.00} = 1.303 \text{ mm Hg/yr}$$

$$a = \bar{Y} - b\bar{X} = 137.2 - (1.303)(52.5) = 68.79 \text{ mm Hg}$$

Therefore, the least squares regression line is $\hat{Y}_{ij} = 68.79 + 1.303X_{ij}$.

Comparing two slopes

(Zar 1999)

$$t = \frac{b_1 - b_2}{s_{b_1 - b_2}}$$

$$s_{b_1 - b_2} = \sqrt{\frac{(s^2_{Y\cdot X})_p}{(\sum \chi_1^2)} + \frac{(s^2_{Y\cdot X})_p}{(\sum \chi_2^2)}}$$

$$(s^2_{Y\cdot X})_p = \frac{(\text{residual SS})_1 + (\text{residual SS})_2}{(\text{residual DF})_1 + (\text{residual DF})_2}$$

$$\text{DF} = n_1 + n_2 - 4$$

Comparing more than two slopes and elevations (Zar 1999)

	$\sum x^2$	$\sum xy$	$\sum y^2$	Residual SS	Residual DF
Regression 1	A_1	B_1	C_1	$SS_1 = C_1 - \frac{B_1^2}{A_1}$	$DF_1 = n_1 - 2$
Regression 2	A_2	B_2	C_2	$SS_2 = C_2 - \frac{B_2^2}{A_2}$	$DF_2 = n_2 - 2$
$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$
Regression k	A_k	B_k	C_k	$SS_k = C_k - \frac{B_k^2}{A_k}$	$DF_k = n_k - 2$
"Pooled" regression				$SS_p = \sum_{i=1}^{k} SS_i$	$DF_p = \sum_{i=1}^{k} (n_i - 2) = \sum_{i=1}^{k} n - 2k$
"Common" regression	$A_c = \sum_{i=1}^{k} A_i$	$B_c = \sum_{i=1}^{k} B_i$	$C_c = \sum_{i=1}^{k} C_i$	$SS_c = C_c - \frac{B_c^2}{A_c}$	$DF_c = \sum_{i=1}^{k} n_i - k - 1$
"Total" regression*	A_t	B_t	C_t	$SS_t = C_t - \frac{B_t^2}{A_t}$	$DF_t = \sum_{i=1}^{k} n_i - 2$

Test differences among slopes

$$F = \frac{\frac{SS_c - SS_p}{k-1}}{\frac{SS_p}{DF_p}}$$

Test differences among elevations

$$F = \frac{\frac{SS_t - SS_c}{k-1}}{\frac{SS_c}{DF_c}}$$

R code and results

```
trees
   Girth Height Volume
1    8.3     70   10.3
2    8.6     65   10.3
3    8.8     63   10.2
4   10.5     72   16.4
5   10.7     81   18.8
6   10.8     83   19.7
7   11.0     66   15.6
8   11.0     75   18.2
9   11.1     80   22.6
10  11.2     75   19.9
11  11.3     79   24.2
12  11.4     76   21.0
13  11.4     76   21.4
14  11.7     69   21.3
15  12.0     75   19.1
16  12.9     74   22.2
17  12.9     85   33.8
18  13.3     86   27.4
19  13.7     71   25.7
20  13.8     64   24.9
21  14.0     78   34.5
22  14.2     80   31.7
23  14.5     74   36.3
24  16.0     72   38.3
25  16.3     77   42.6
26  17.3     81   55.4
27  17.5     82   55.7
28  17.9     80   58.3
29  18.0     80   51.5
30  18.0     80   51.0
31  20.6     87   77.0
```

```
reg.tree <- lm(Volume~Height, data=trees)
reg.tree
```

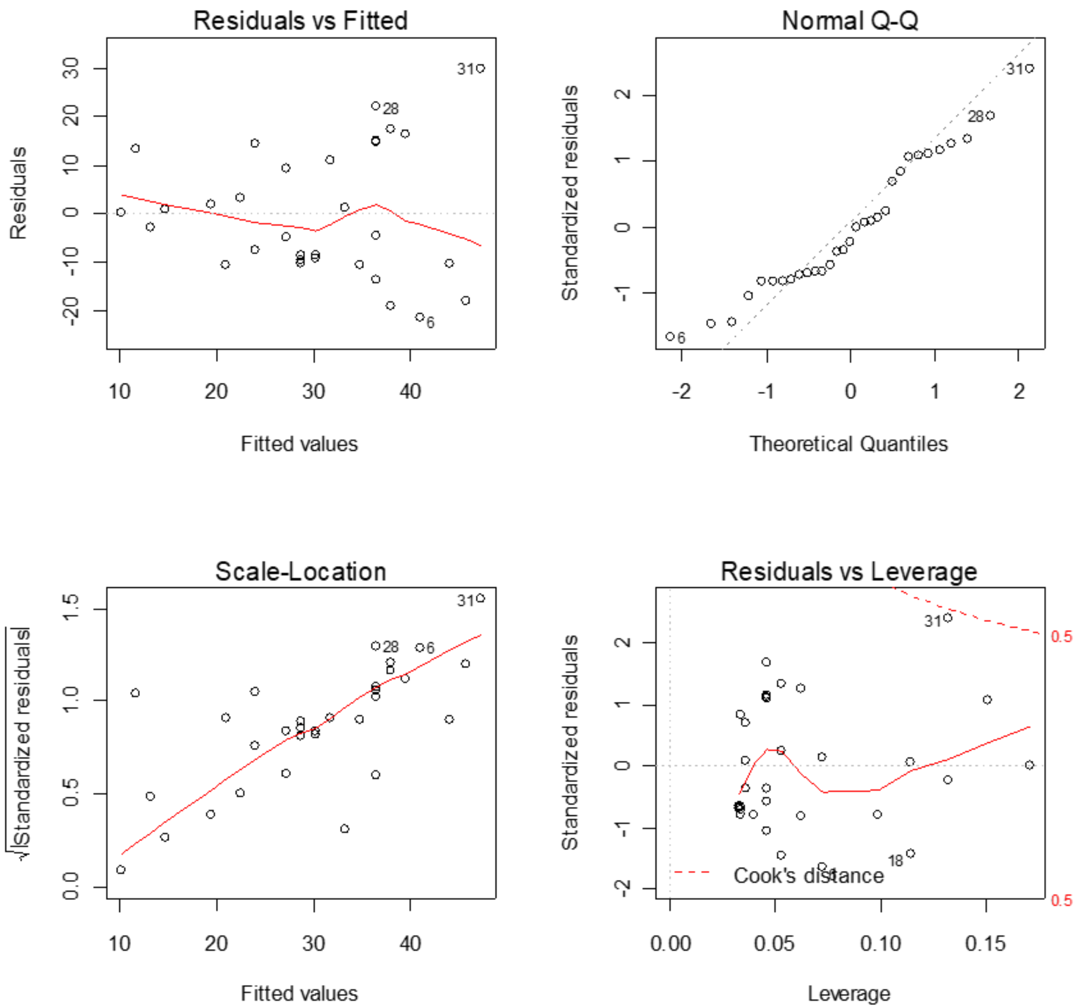
```
Call:
lm(formula = Volume ~ Height, data = trees)

Coefficients:
(Intercept)       Height
    -87.124        1.543
```

```
summary(reg.tree)
```

```
Call:
lm(formula = Volume ~ Height, data = trees)

Residuals:
    Min      1Q  Median      3Q     Max
-21.274  -9.894  -2.894  12.067  29.852

Coefficients:
            Estimate  Std. Error  t value   Pr(>|t|)
(Intercept) -87.1236   29.2731    -2.976   0.005835 **
Height        1.5433    0.3839     4.021   0.000378 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 13.4 on 29 degrees of freedom
Multiple R-squared: 0.3579,     Adjusted R-squared: 0.3358
F-statistic: 16.16 on 1 and 29 DF,  p-value: 0.0003784
```

Check the model

```
par(mfrow=c(2,2))
plot(lm(Volume~Height, data=trees))
```



Check the model

```
library(car)
fit = lm(Girth ~ Height, data = trees)

# Computes residual autocorrelations and generalized Durbin-Watson statistics
# and their bootstrapped p-values
durbinWatsonTest(fit) # check independence
# P < 0.05, autocorrelation exists.

# component + residual plots (also called partial-residual plots) for linear
# and generalized linear models
crPlots(fit) # check linearity
# the red line (regression) and green line (residual) match well, the linearity is good.

# Score Test for Non-Constant Error Variance
ncvTest(fit)
# p = 0.15, error variance is homogeneous
```

Simple linear correlation

Simple linear correlation

1. Answer **How Strong** is the linear relationship between 2 variables?

2. Coefficient of correlation

   Values Range from -1 to +1

   Measures Degree of Association

3. Used mainly for understanding

Pearson product moment coefficient of correlation, *r* (Pearson correlation coefficient)

$$r = \frac{\sum_{i=1}^{n}\left(X_i - \bar{X}\right)\left(Y_i - \bar{Y}\right)}{\sqrt{\sum_{i=1}^{n}\left(X_i - \bar{X}\right)^2} \cdot \sqrt{\sum_{i=1}^{n}\left(Y_i - \bar{Y}\right)^2}}$$

```
# correlation coefficient and p value
cor(X, Y, use = 'pairwise.complete.obs')
cor.test(X, Y, alternative = c("two.sided"), method = c("pearson"))$p.value

# correlation coefficient
r = SSXY/(SSX*SSY)^.5
```

Understanding Pearson correlation coefficient

$$r = \frac{\sum_{i=1}^{n}(X_i - \bar{X})(Y_i - \bar{Y})}{\sqrt{\sum_{i=1}^{n}(X_i - \bar{X})^2} \cdot \sqrt{\sum_{i=1}^{n}(Y_i - \bar{Y})^2}}$$

$$\sum_{i=1}^{n}(X_i - \bar{X})(Y_i - \bar{Y}) = covariance \times n$$

$$\sum_{i=1}^{n}(X_i - \bar{X})^2 = SSX \qquad \sum_{i=1}^{n}(Y_i - \bar{Y})^2 = SSY$$

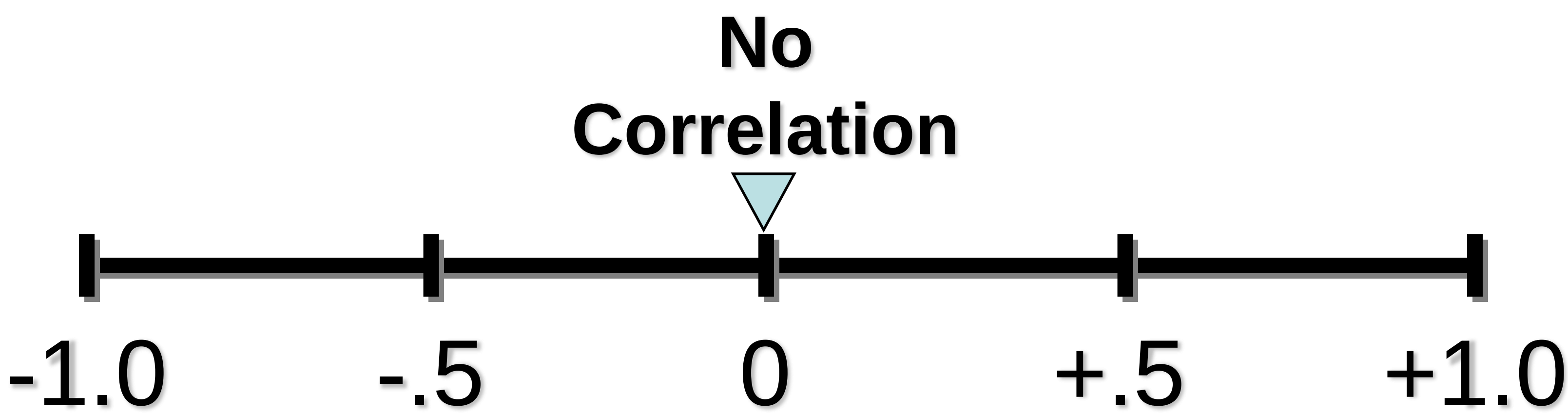
Understanding Pearson correlation coefficient

When $(X_i - \bar{X}) = (Y_i - \bar{Y})$

$(X_i - \bar{X})(Y_i - \bar{Y})$ maximized

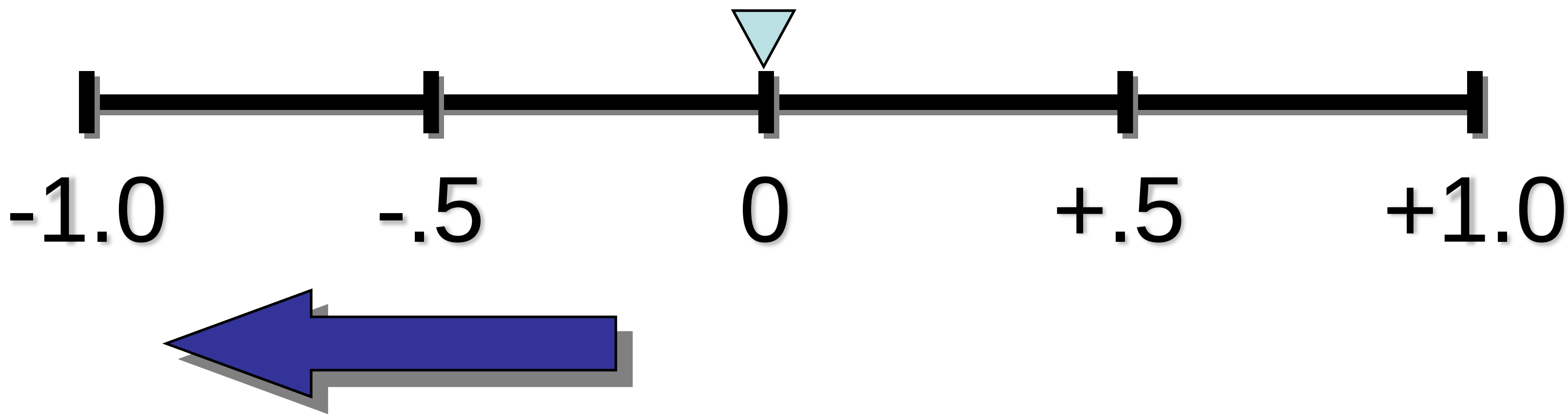
ΔX	ΔY	ΔX+ΔY	ΔXΔY
3	3	6	9
2	4	6	8
1	5	6	5
0.5	5.5	6	2.75

Coefficient of Correlation Values



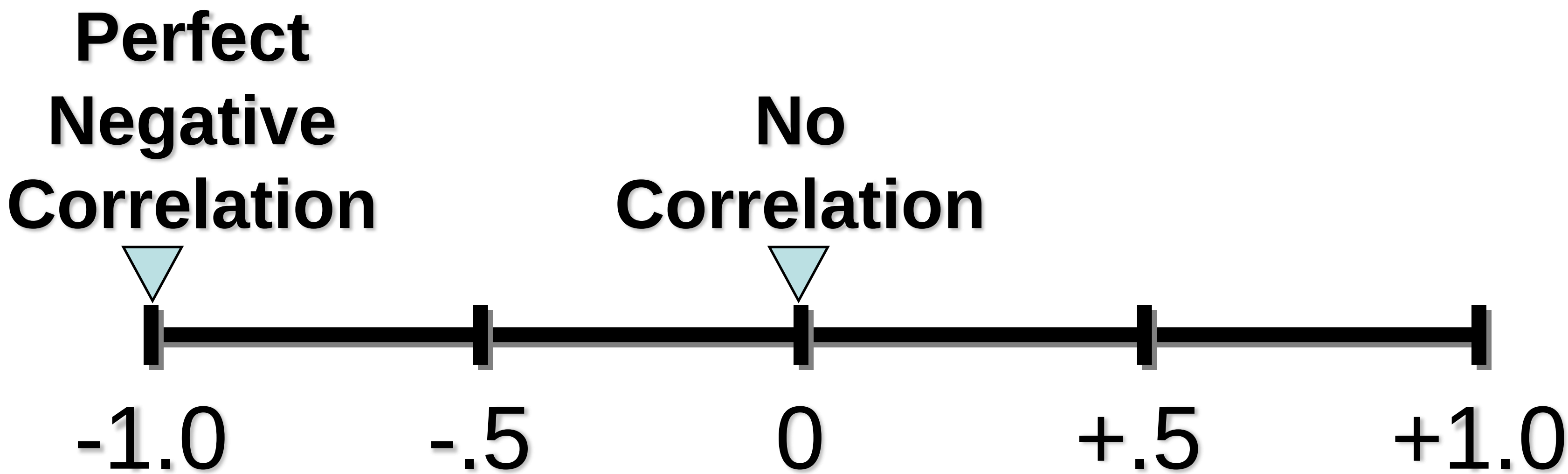
Coefficient of Correlation Values

No Correlation

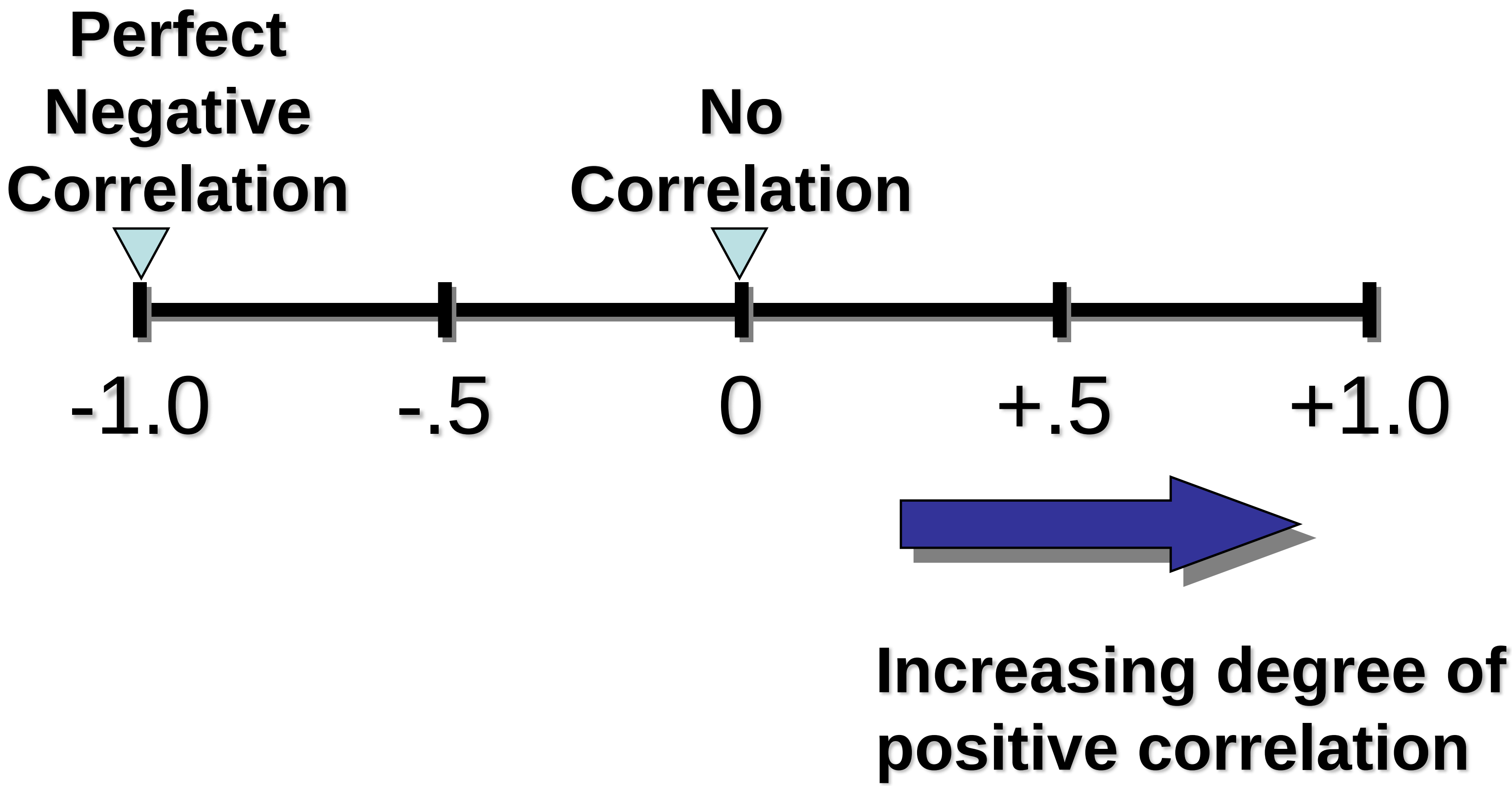


Increasing degree of negative correlation

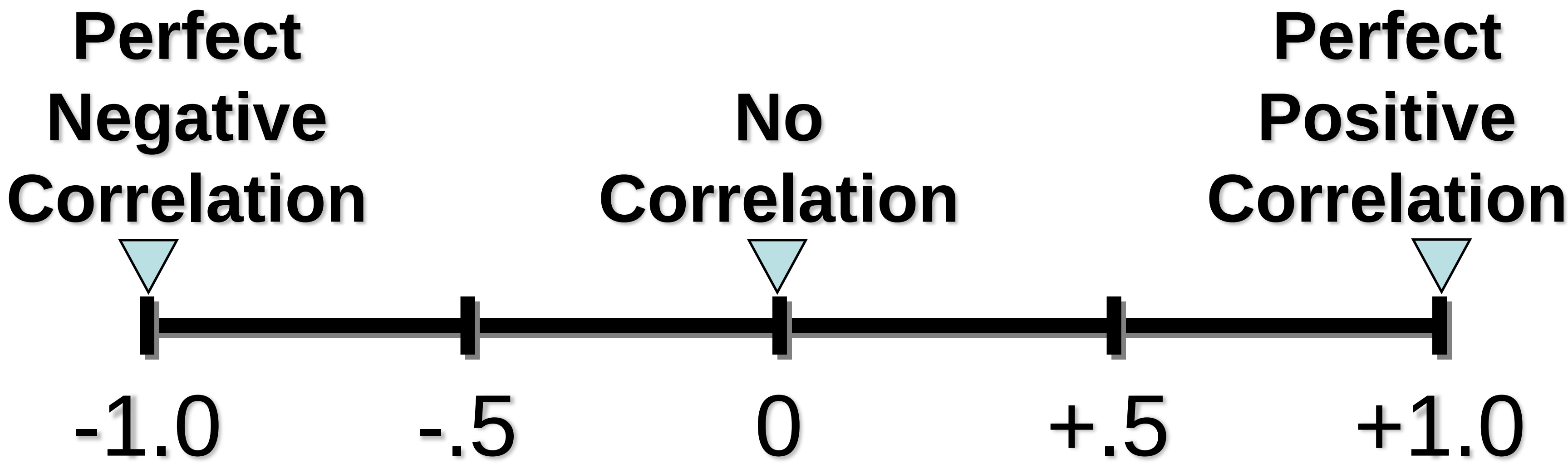
Coefficient of Correlation Values



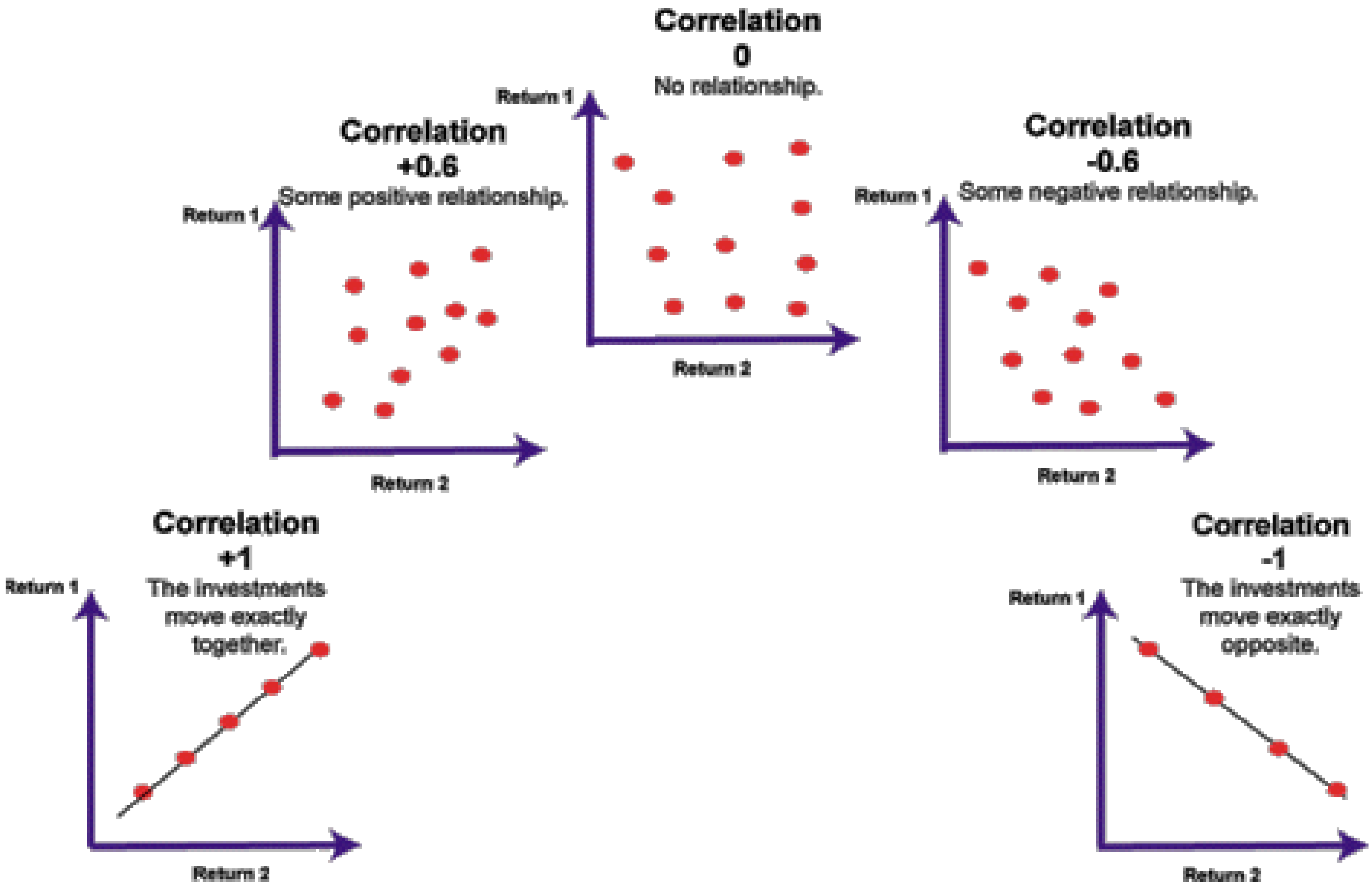
Coefficient of Correlation Values



Coefficient of Correlation Values



Coefficient of Correlation



Interpretation of Correlation

Correlations

- From 0 to 0.25 (-0.25) = little or no relationship;
- From 0.25 to 0.50 (-0.25 to 0.50) = fair degree of relationship;
- From 0.50 to 0.75 (-0.50 to -0.75) = moderate to good relationship;
- Greater than 0.75 (or -0.75) = very good to excellent relationship.

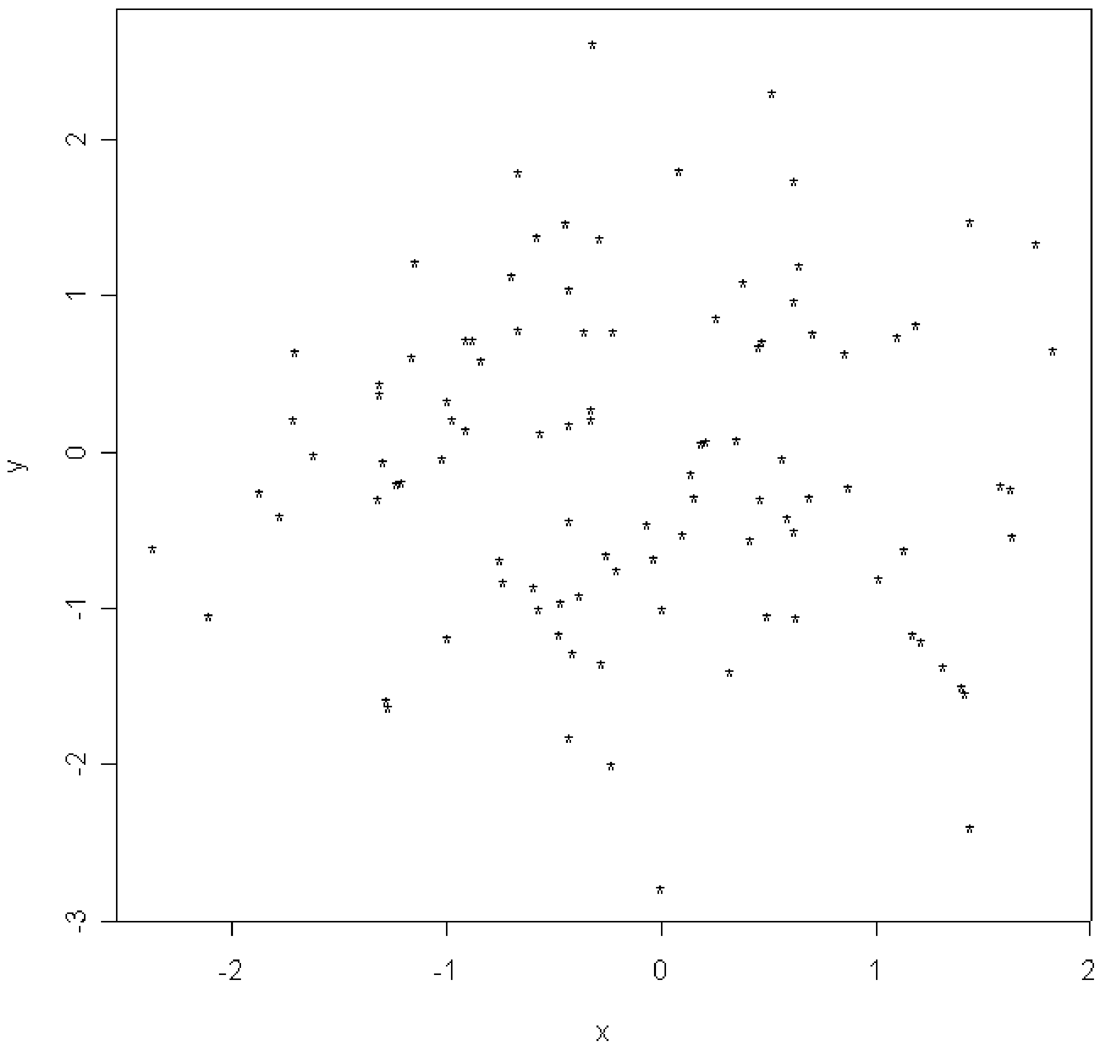
Limitations of the correlation coefficient

- Though ***r*** measures how closely the two variables approximate a straight line, it does not validly measures the strength of nonlinear relationship

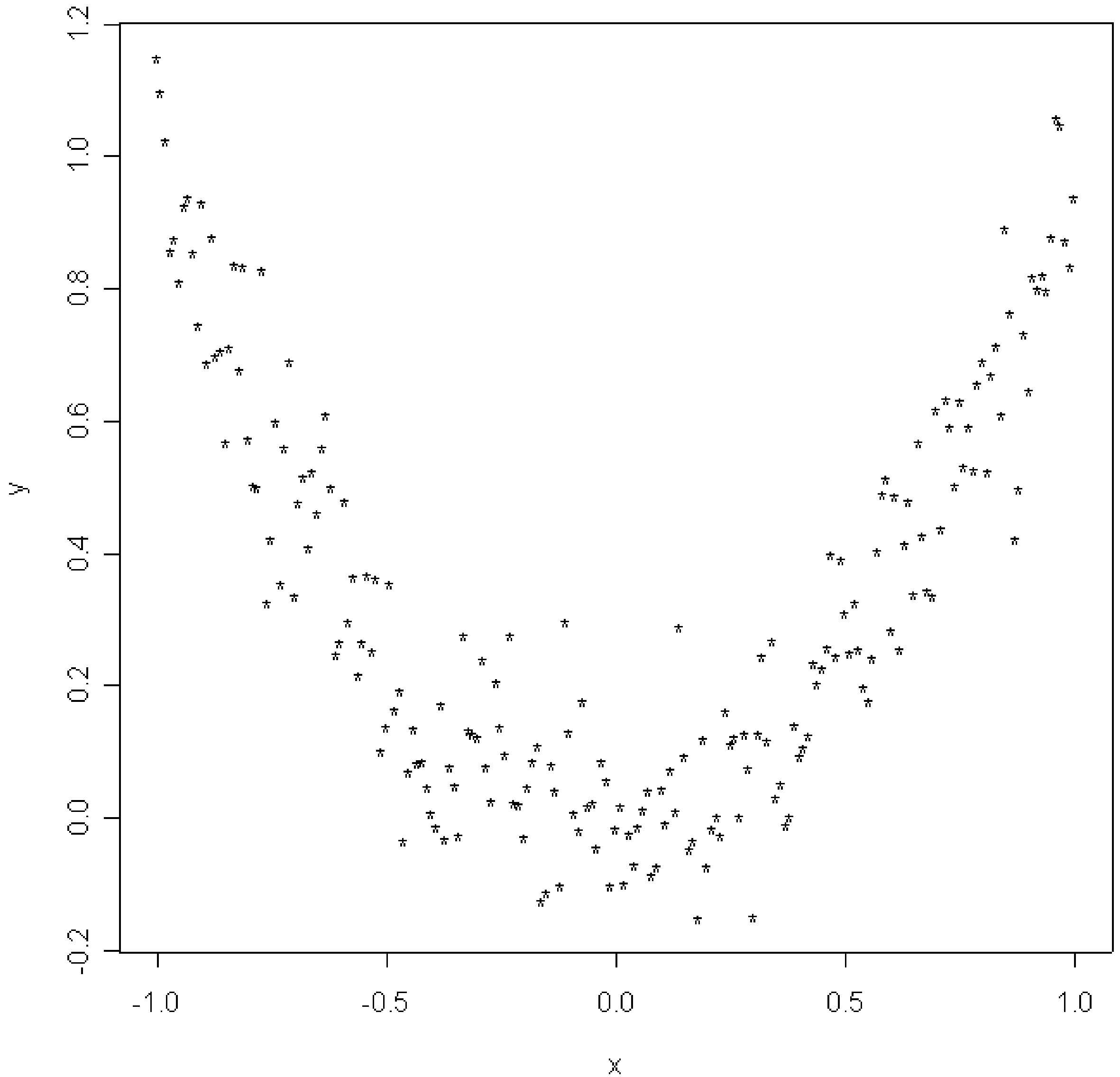
- When the sample size, n, is small we also have to be careful with the reliability of the correlation

- Outliers could have a marked effect on ***r***

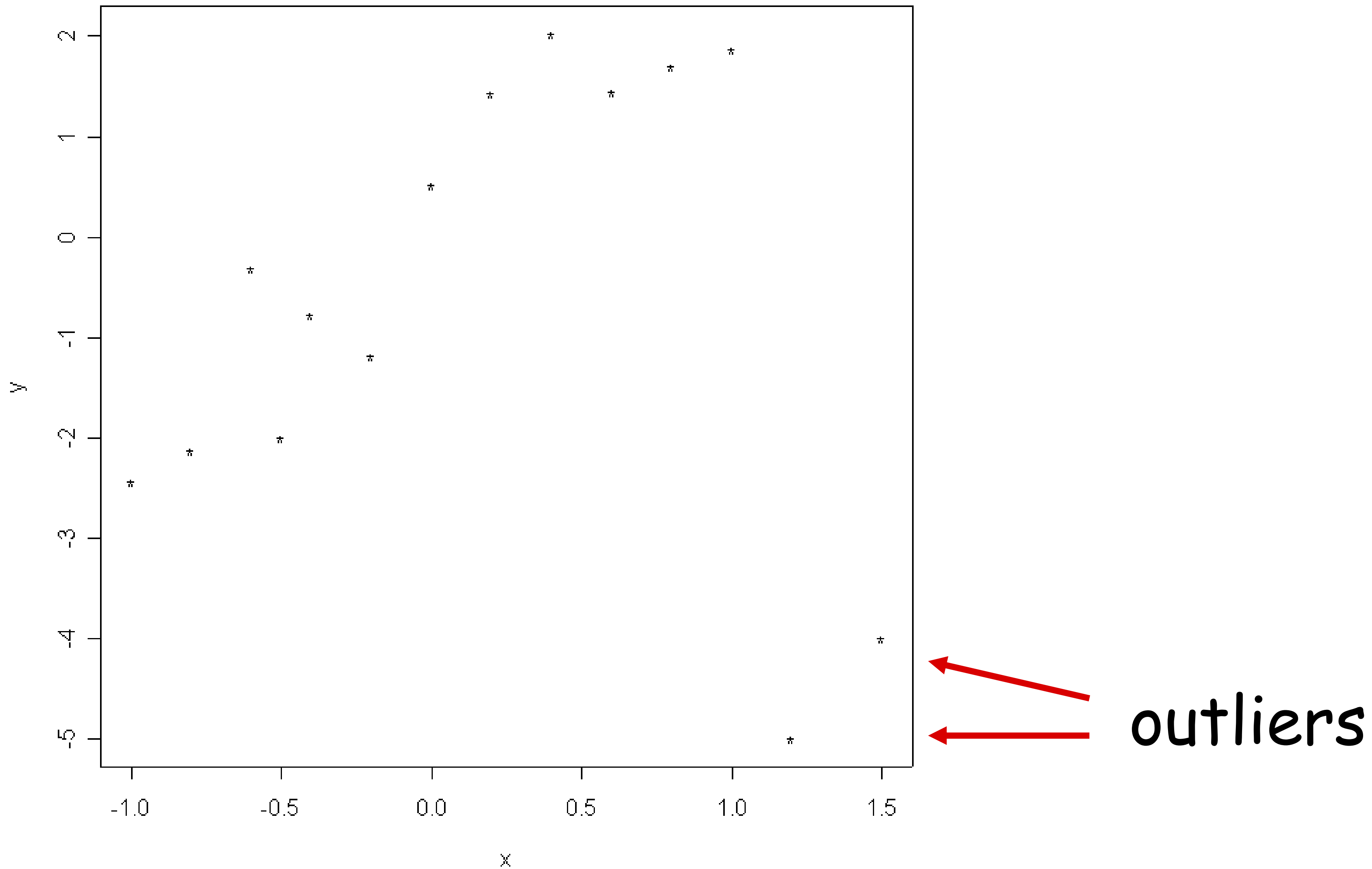
$r \approx 0$: random scatter



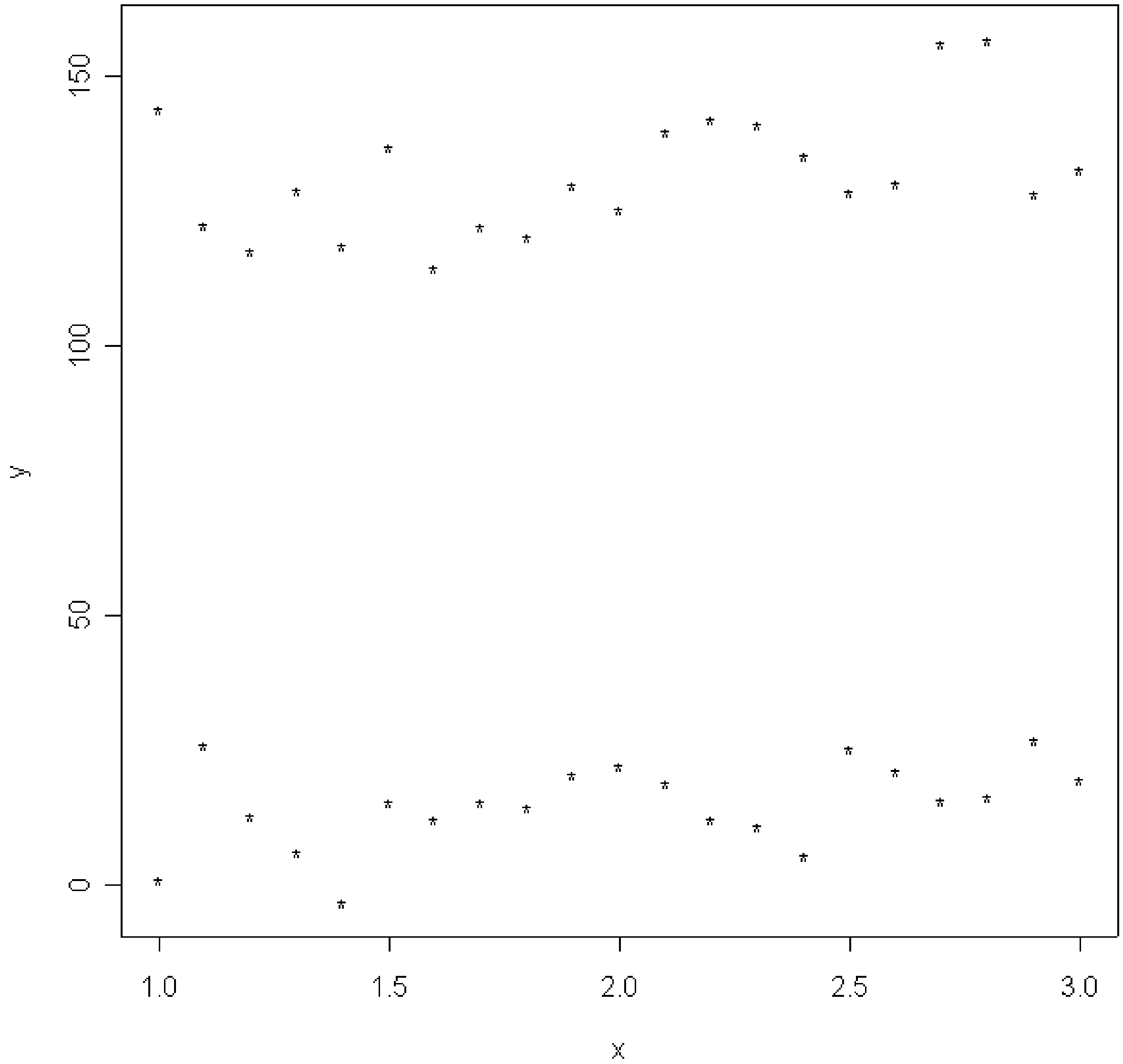
$r \approx 0$: curved relation



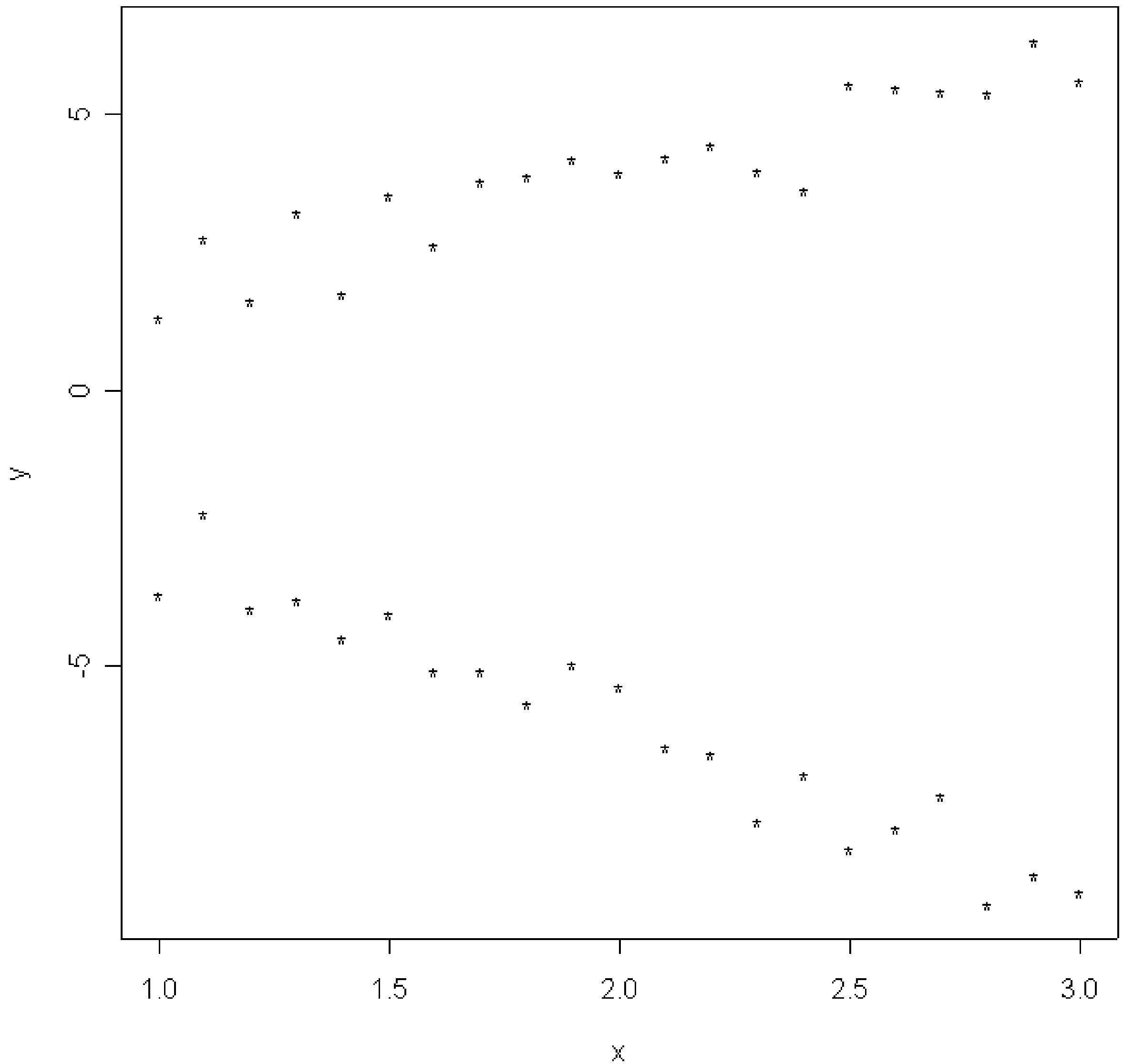
$r \approx 0$: outliers



$r \approx 0$: parallel lines



$r \approx 0$: different linear trends



Reading Correlation Matrix

Correlations[a]

		Total ball toss points	Distance from target	Time spun before throwing	Aiming accuracy	Manual dexterity	College grade point avg	Confidence for task
Total ball toss points	Pearson Correlation	1	-.904*	-.582	.628	.821*	-.037	-.502
	Sig. (2-tailed)	.	.013	.226	.181	.045	.945	.310
	N	6	6	6	6	6	6	6
Distance from target	Pearson Correlation	-.904*	1	.279	-.653	-.883*	.228	.522
	Sig. (2-tailed)	.013	.	.592	.159	.020	.664	.288
	N	6	6	6	6	6	6	6
Time spun before throwing	Pearson Correlation	-.582	.279	1	-.390	-.248	-.087	.267
	Sig. (2-tailed)	.226	.592	.	.445	.635	.869	.609
	N	6	6	6	6	6	6	6
Aiming accuracy	Pearson Correlation	.628	-.653	-.390	1	.758	-.546	-.250
	Sig. (2-tailed)	.181	.159	.445	.	.081	.262	.633
	N	6	6	6	6	6	6	6
Manual dexterity	Pearson Correlation	.821*	-.883*	-.248	.758	1	-.553	-.101
	Sig. (2-tailed)	.045	.020	.635	.081	.	.255	.848
	N	6	6	6	6	6	6	6
College grade point avg	Pearson Correlation	-.037	.228	-.087	-.546	-.553	1	-.524
	Sig. (2-tailed)	.945	.664	.869	.262	.255	.	.286
	N	6	6	6	6	6	6	6
Confidence for task	Pearson Correlation	-.502	.522	.267	-.250	-.101	-.524	1
	Sig. (2-tailed)	.310	.288	.609	.633	.848	.286	.
	N	6	6	6	6	6	6	6

*. Correlation is significant at the 0.05 level (2-tailed).

a. Day sample collected = Tuesday

Distance from target	Pearson Correlation	-.904*
	Sig. (2-tailed)	.013
	N	6

$r = -.904$

p = .013 -- Probability of getting a correlation this size by sheer chance. **Reject Ho if p ≤ .05.**

sample size

$r(4) = -.904, p \leq .05$

Hypothesis test about correlation coefficient

$$t = \frac{r}{s_r}$$

$$s_r = \sqrt{\frac{1 - r^2}{n - 2}}$$

df = n-2

$$t_{\alpha(2),\, n\text{-}2}$$

Hypothesis test about correlation coefficient

$$F = \frac{1+|r|}{1-|r|}$$

(Cacoullos, 1965)

$df_n = df_d = n-2$

Hypothesis test about correlation coefficient

(Zar 1999)

EXAMPLE 19.1b Testing H_0: $\rho = 0$ vs. H_A: $\rho \neq 0$. The data are those of Example 19.1a.

From Example 19.1a: $r = 0.870$.

To test H_0: $\rho = 0$ vs. H_A: $\rho \neq 0$:

$$\text{standard error of } r = s_r = \sqrt{\frac{1 - r^2}{n - 2}} = \sqrt{\frac{1 - (0.870)^2}{12 - 2}} = 0.156$$

$$t = \frac{r}{s_r} = \frac{0.870}{0.156} = 5.58$$

$$t_{0.05(2),10} = 2.228$$

Therefore, reject H_0.

$$P < 0.001 \quad [P = 0.00012]$$

Or:

$$F = \frac{1 + |r|}{1 - |r|} = \frac{1.870}{0.130} = 14.4$$

$$F_{0.05(2),10,10} = 3.72$$

Therefore, reject H_0.

$$P < 0.001 \quad [P = 0.00014]$$

Fisher's *z* transformation

(Zar 1999)

When r is not normal $$z = 0.5\ln\left(\frac{1+r}{1-r}\right) \qquad \sigma_z = \sqrt{\frac{1}{n-3}}$$

EXAMPLE 19.2 Testing H_0: $\rho = \rho_0$, where $\rho_0 \neq 0$.

$r = 0.870$

$n = 12$

H_0: $\rho = 0.750$; H_A: $\rho \neq 0.750$.

$z = 1.3331$

$\zeta_0 = 0.9730$

$$Z = \frac{z - \zeta_0}{\sqrt{\frac{1}{n-3}}} = \frac{1.3331 - 0.9730}{\sqrt{\frac{1}{9}}} = \frac{0.3601}{0.3333} = 1.0803$$

$Z_{0.05(2)} = t_{0.05(2),\infty} = 1.960$

Therefore, do not reject H_0.

$0.20 < P < 0.50$ $[P = 0.28]$

Power and sample size in correlation

$$Z_{\beta(1)} = (z - z_{\alpha})\sqrt{n-3}$$

(Cohen 1988)

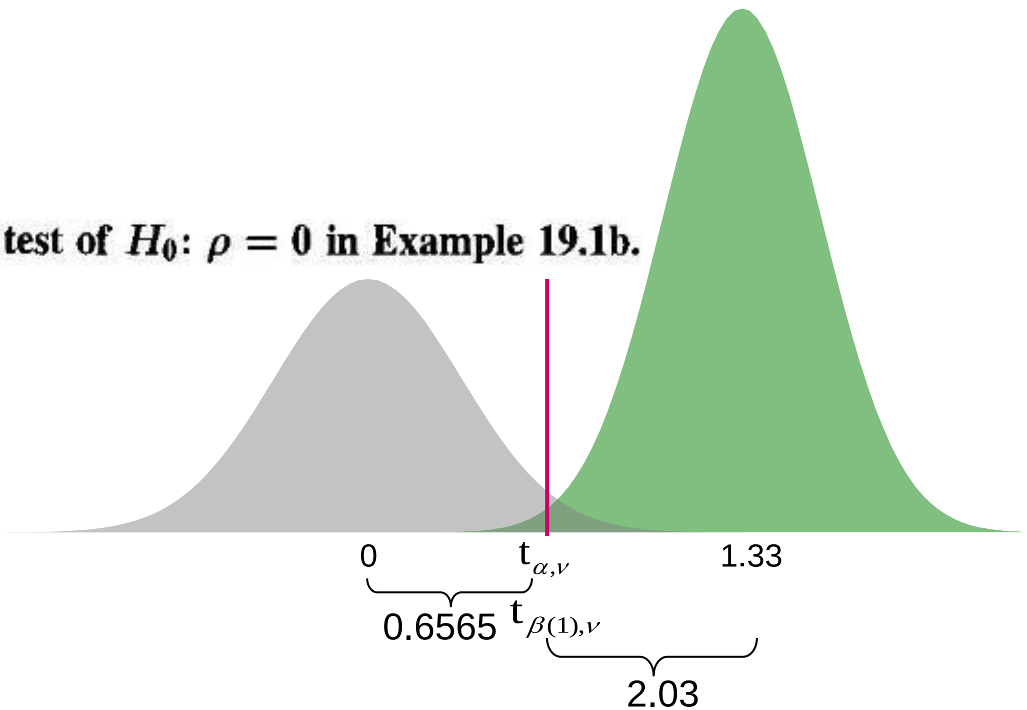
EXAMPLE 19.4 Determination of power of the test of H_0: $\rho = 0$ in Example 19.1b.

$n = 12;\ \nu = 10$

$r = 0.870$, so $z = 1.3331$

$r_{0.05(2),10} = 0.576$, so $z_{0.05} = 0.6565$

$$Z_{\beta(1)} = (1.3331 - 0.6565)\sqrt{12-3}$$
$$= 2.03$$



From Appendix Table B.2, $P(Z \geq 2.03) = 0.0212 = \beta$. Therefore, the power of the test is $1 - \beta = 0.98$.

Sample size in correlation

$$n = \left(\frac{Z_{\beta(1)} + Z_\alpha}{\zeta_0}\right)^2 + 3,$$

where ζ_0 is the Fisher transformation of the ρ_0 specified, and the significance level, can be either one-tailed or two-tailed. This procedure is shown in Example 19.5.

EXAMPLE 19.5 Determination of required sample size in testing H_0: $\rho = 0$.

We desire to reject H_0: $\rho = 0$ 99% of the time when $|\rho| \geq 0.5$ and the hypothesis is tested at the 0.05 level of significance. Therefore, $\beta(1) = 0.01$ and (from the last line of Appendix Table B.3) and $Z_{\beta(1)} = 2.3263$; $\alpha(2) = 0.05$ and $Z_{\alpha(2)} = 1.9600$; and, for $r = 0.5$, $z = 0.5493$.

Then,

$$n = \left(\frac{2.3263 + 1.9600}{0.5493}\right)^2 + 3 = 63.9,$$

so a sample of size at least 64 should be used.

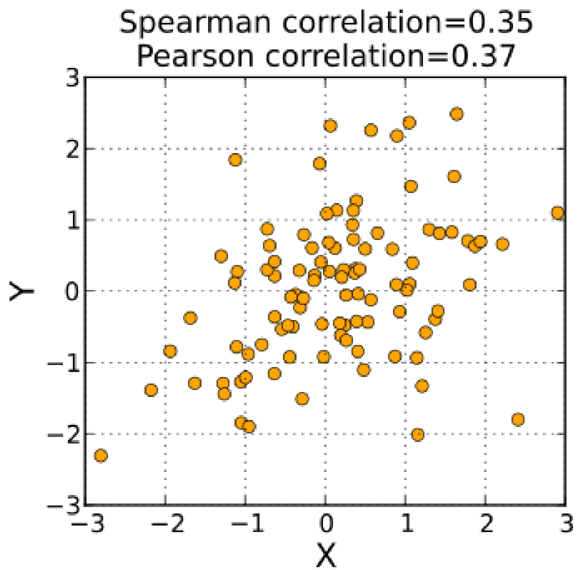
Types of Coefficients

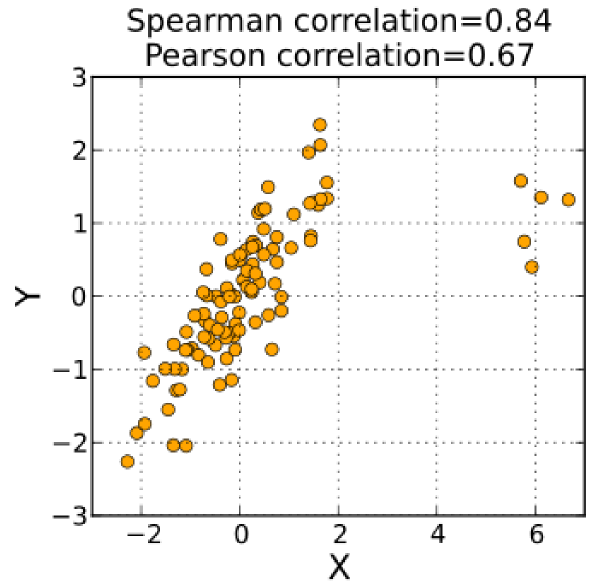
Type of Data	Correlation Coefficient
Continuous v. Continuous	Pearson’s r
Continuous v. Ordinal	Jaspen’s Multiserial Coefficient (M)
Ordinal v. Ordinal	Spearman’s ρ (Rho) Kendall’s τ (Tau)

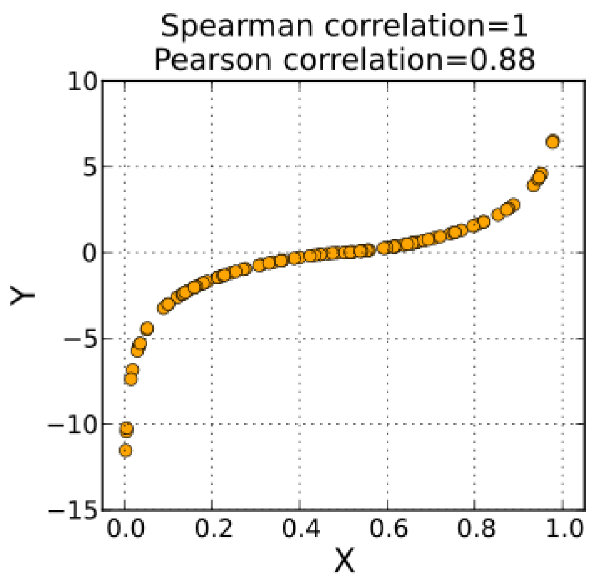
Spearman correlation coefficient

```
head(mtcars)
plot(mtcars$wt, mtcars$hp)
# Pearson correlation coefficient
cor(mtcars$wt, mtcars$hp) # 0.66

wt.rank = rank(mtcars$wt)
hp.rank = rank(mtcars$hp)
# Spearman correlation coefficient
cor(wt.rank, hp.rank) # 0.77
```

Pearson's r vs. Spearman's ρ








```
X = seq(0.5* pi, 1.5*pi, length=100)
Y = 1 - sin(X)
plot(Y, X)
cor(X, Y) # 0.99
cor(rank(X),rank(Y)) # 1
```

http://en.wikipedia.org/wiki/Spearman%27s_rank_correlation_coefficient

Correlations with significance levels

	mpg	cyl	disp	hp	drat	wt	qsec	vs	am	gear	carb
mpg	1	-0.85	-0.85	-0.78	0.68	-0.87	0.42	0.66	0.6	0.48	-0.55
cyl	-0.85	1	0.9	0.83	-0.7	0.78	-0.59	-0.81	-0.52	-0.49	0.53
disp	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	-0.56	0.39
hp	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	-0.13	0.75
drat	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	0.7	-0.09
wt	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	-0.58	0.43
qsec	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	-0.21	-0.66
vs	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	0.21	-0.57
am	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	0.79	0.06
gear	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	1	0.27
carb	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	0.27	1

```
# Correlations with significance levels
library(Hmisc)
# mtcars is a dataframe
head(mtcars)
rcorr(as.matrix(mtcars), type="pearson") #parametric
rcorr(as.matrix(mtcars), type="spearman")#nonparametric
```

n= 32

P

	mpg	cyl	disp	hp	drat	wt	qsec	vs	am	gear	carb
mpg		0	0	0	0	0	0.0171	0	0.0003	0.0054	0.0011
cyl	0		0	0	0	0	0.0004	0	0.0022	0.0042	0.0019
disp	0	0		0	0	0	0.0131	0	0.0004	0.001	0.0253
hp	0	0	0		0.01	0	0	0	0.1798	0.493	0
drat	0	0	0	0.01		0	0.6196	0.0117	0	0	0.6212
wt	0	0	0	0	0		0.3389	0.001	0	0.0005	0.0146
qsec	0.0171	0.0004	0.0131	0	0.6196	0.3389		0	0.2057	0.2425	0
vs	0	0	0	0	0.0117	0.001	0		0.357	0.2579	0.0007
am	0.0003	0.0022	0.0004	0.1798	0	0	0.2057	0.357		0	0.7545
gear	0.0054	0.0042	0.001	0.493	0	0.0005	0.2425	0.2579	0		0.129
carb	0.0011	0.0019	0.0253	0	0.6212	0.0146	0	0.0007	0.7545	0.129	

Simple linear regression and correlation

$$r^2 = \frac{\mathrm{SP}^2}{\mathrm{SS_x SS_y}} = \frac{\mathrm{SP}}{\mathrm{SS_x}} \cdot \frac{\mathrm{SP}}{\mathrm{SS_y}} = \mathrm{b_{y/x} b_{x/y}}$$

$$\beta = \frac{\sum \mathrm{xy}}{\sum \mathrm{x}^2}$$

Correlation for categorical variables (contingency table)

Treat.2	Treat.1 +	Treat.1 −
+	*a*	*b*
−	*c*	*d*

$$r_n = \frac{ad - bc}{\sqrt{(a+b)(c+d)(a+c)(b+d)}}$$

$$-1 \le r_n \le +1$$

Colignatus, Thomas. 2007. Correlation and regression in contingency tables. A measure of association or correlation in nominal data (contingency tables), using determinants. MPRA Paper No. 3660. http://mpra.ub.uni-muenchen.de/3660/

Assignment: simple linear regression

Tasks:

- Describe your question: dependence of one variable to another
- Point out hypothesis Ho and Ha
- Develop your dataset
- List the following input and output contents
  - R Commands
  - R Output
  - R Plot - Y*X
  - R Plot - RESIDUAL*PREDICTED
  - R Plot - RESIDUAL*X
- Check assumptions
- Give r square value
- Write conclusion

R

```
head(trees)
reg.tree = lm(Volume~Height, data=trees)
summary(reg.tree)
plot(trees$Height, trees$Volume) # x-y plot
plot(reg.tree$fitted, reg.tree$resid) # check homogeneity
shapiro.test(reg.tree$resid) # check normality
summary(lm(Volume~Height, trees))$r.squared # R square
```

	Girth	Height	Volume
1	8.3	70	10.3
2	8.6	65	10.3
3	8.8	63	10.2
4	10.5	72	16.4
5	10.7	81	18.8
6	10.8	83	19.7

Assignment 5 (option 2)：Correlation

Tasks:

Develop a CORRELATION experimental design. Generate your own data and FORMALIZE your hypotheses.

Define 4-6 variables (all at ratio scale)

Print out the correlation matrix

Highlight the pairs with significant p values of 0.05 and 0.01 separately

Change data to reach the following results:

- same r, different p value (one significant, one not significant)
- same p value, different r

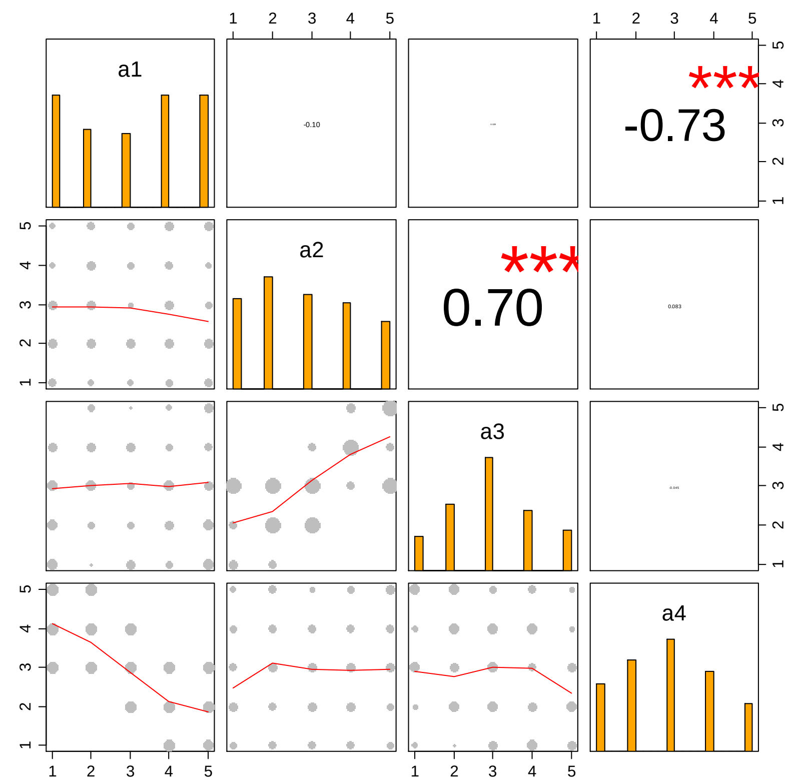
List the source data and the r and p values.

R code for producing a correlation scatter-plot matrix

```
## put (absolute) correlations on the upper panels,
## with size proportional to the correlations.
panel.cor <- function(x, y, digits=2, prefix="", cex.cor, ...)
{
  usr <- par("usr"); on.exit(par(usr))
  par(usr = c(0, 1, 0, 1))
  r <- abs(cor(x, y))
  txt <- format(c(r, 0.123456789), digits=digits)[1]
  txt <- paste(prefix, txt, sep="")
  if(missing(cex.cor)) cex.cor <- 0.8/strwidth(txt)
  text(0.5, 0.5, txt, cex = cex.cor * r)
}
pairs(sheds[,4:13], lower.panel = panel.smooth, upper.panel = panel.cor)
```

R code for producing a correlation scatter-plot matrix for ordered-categorical data

```
panel.cor.ordered.categorical <- function(x, y, digits=2, prefix="", cex.cor) {
   usr <- par("usr"); on.exit(par(usr))
   par(usr = c(0, 1, 0, 1))

   r <- abs(cor(x, y, method = "spearman")) # notive we use spearman, non parametric correlation here
   r.no.abs <- cor(x, y, method = "spearman")

   txt <- format(c(r.no.abs , 0.123456789), digits=digits)[1]
   txt <- paste(prefix, txt, sep="")
   if(missing(cex.cor)) cex <- 0.8/strwidth(txt)

   test <- cor.test(x,y, method = "spearman")
   # borrowed from printCoefmat
   Signif <- symnum(test$p.value, corr = FALSE, na = FALSE,
              cutpoints = c(0, 0.001, 0.01, 0.05, 0.1, 1),
              symbols = c("***", "**", "*", ".", " "))

   text(0.5, 0.5, txt, cex = cex * r)
   text(.8, .8, Signif, cex=cex, col=2)
}

panel.smooth.ordered.categorical <- function (x, y, col = par("col"), bg = NA, pch = par("pch"),
                             cex = 1, col.smooth = "red", span = 2/3, iter = 3,
                             point.size.rescale = 1.5, ...)
{
   #require(colorspace)
   require(reshape)
   z <- merge(data.frame(x,y), melt(table(x ,y)),sort =F)$value
   #the.col <- heat_hcl(length(x))[z]
   z <- point.size.rescale*z/ (length(x)) # notice how we rescale the dots accourding to the maximum z could have gotten

   symbols( x, y,  circles = z,#rep(0.1, length(x)), #sample(1:2, length(x), replace = T) ,
                             inches=F, bg= "grey",#the.col ,
                             g = bg, add = T)

   # points(x, y, pch = pch, col = col, bg = bg, cex = cex)
   ok <- is.finite(x) & is.finite(y)
   if (any(ok))
      lines(stats::lowess(x[ok], y[ok], f = span, iter = iter),
         col = col.smooth, ...)
}

panel.hist <- function(x, ...) {
   usr <- par("usr"); on.exit(par(usr))
   par(usr = c(usr[1:2], 0, 1.5) )
   h <- hist(x, plot = FALSE, br = 20)
   breaks <- h$breaks; nB <- length(breaks)
   y <- h$counts; y <- y/max(y)
   rect(breaks[-nB], 0, breaks[-1], y, col="orange", ...)
}

pairs.ordered.categorical <- function(xx,...) {
                             pairs(xx ,
                             diag.panel = panel.hist ,
                             lower.panel=panel.smooth.ordered.categorical,
                             upper.panel=panel.cor.ordered.categorical,
                             cex.labels = 1.5, ...)
                             }
# Example
set.seed(666)
a1 <- sample(1:5, 100, replace = T)
a2 <- sample(1:5, 100, replace = T)
a3 <- round(jitter(a2, 7) )
  a3[a3 < 1 | a3 > 5] <- 3
a4 <- 6-round(jitter(a1, 7) )
  a4[a4 < 1 | a4 > 5] <- 3
 aa <- data.frame(a1,a2,a3, a4)
 require(reshape)
 # plotting :)
pairs.ordered.categorical(aa)
```



http://www.r-statistics.com/2010/04/correlation-scatter-plot-matrix-for-ordered-categorical-data/