Von Geraden und Kurven

Regression und Korrelation mit Daten von Schulkindern und zur Holztrocknung

von Norbert Kessel

Regression und Korrelation

Häufig sucht man nach Zusammenhängen zwischen zwei variablen Größen, zum Beispiel der Körpergröße von Menschen und deren Gewicht, oder dem Zusammenhang zwischen dem Durchmesser eines Baumes und seiner Höhe.

Im Hintergrund steht der Gedanke, dass man – wenn eine Beziehung zwischen den beiden Variablen vorliegen sollte – mit nur einer Messung der ersten Variable eine Vorhersage für die zweite Variable machen kann. Das ist bei Bäumen besonders praktisch, da man den Durchmesser leicht, die Höhe aber oft nur mit großem Aufwand ermitteln kann. Man kann mit einem Zusammenhang, den man üblicherweise mit einer Formel beschreibt, eine Voraussage machen, wann ein bestimmter Wert erreicht sein wird, zum Beispiel bei der Trocknung von Holz, hierzu ist unten ein Beispiel enthalten.

Häufig fängt man damit an, dass man Messwert-Paare (z.B. Gewicht und Größe) in ein x-y-Koordinatensystem einzeichnet. Dabei entsteht eine Punktewolke (s. Abbildungen unten). Anschließend werden zwei Rechenverfahren verwendet:

Die **Regressionsanalyse** versucht herauszufinden, *welcher Art der Zusammenhang* zwischen zwei Variablen ist. Im einfachsten Fall ist er linear und lässt sich mit einer einfachen Funktion beschreiben. Anhand dieser Funktion kann man dann in die Punktewolke eine Ausgleichsgerade legen.

Man unterscheidet zwei grundsätzliche Arten von Regressionen:

- lineare Regression (mit den Varianten: einfache lineare R. und multiple lineare R.), dabei wird der Ausgleich mit einer Geraden hergestellt,
- nicht-lineare Regression, dabei nimmt man den Ausgleich mit einer Kurve vor.

Die **Korrelationsanalyse**, die man üblicherweise nach der Regressionsanalyse macht, versucht den *Grad des Zusammenhangs* zu ermitteln, dazu berechnet man den sogeannten Korrelationskoeffizienten.

Nach einer kurzen Einführung zu Datenpaaren und zur linearen Regression folgen Beispiele mit Daten von Schulkindern und zur Holztrocknung.

Die allgemeine Geradengleichung

Zunächst ein Blick zurück in die Schulzeit. Dort gab es Punktepaare, die man zuerst in einer Tabelle gesammelt und dann in ein x-y-Koordinatensystem eingetragen hat, zum Beispiel folgendermaßen:

x	1	2	3	4
y	2	4	6	8

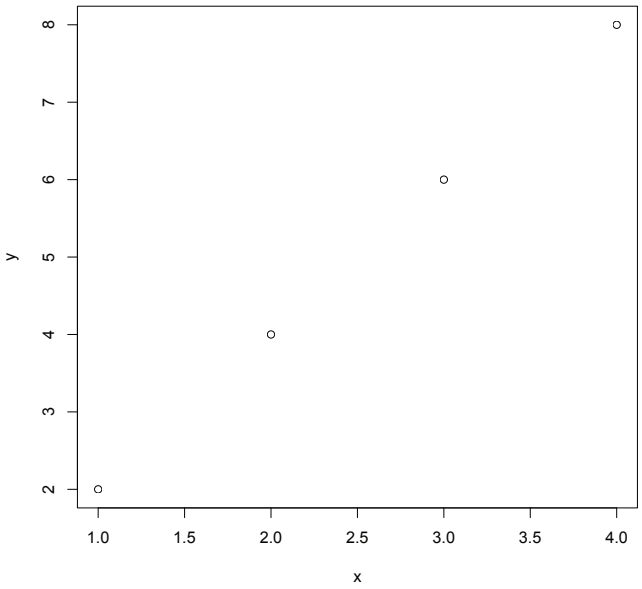
Damit R mit diesen Daten arbeiten kann, müssen sie zuerst in zwei Variablen gespeichert werden (die Namen der Variablen sind frei wählbar):

```
x<-c(1,2,3,4)
y<-c(2,4,6,8)
```

Anschließend können die Punktepaare in ein x-y-Koordinatensystem gezeichnet werden. Die Anweisung:

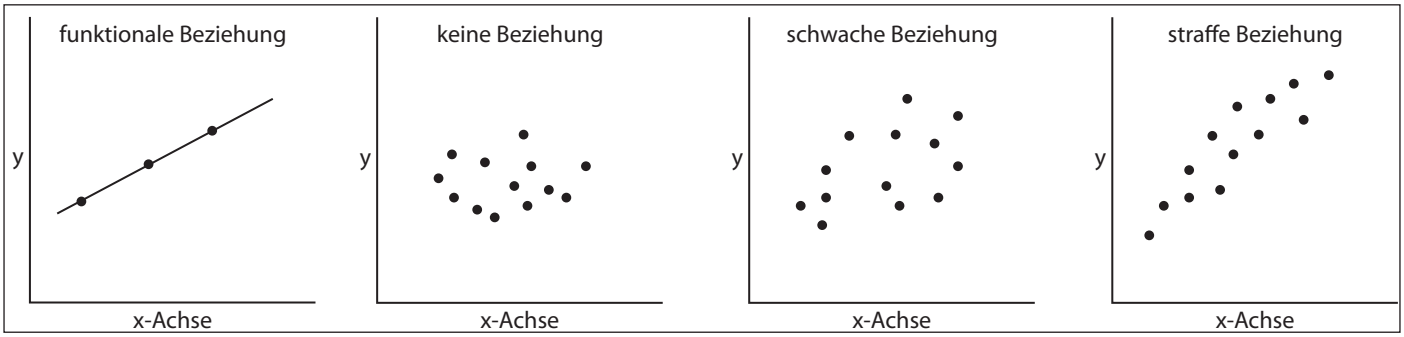
```
plot(x,y)
```

führt zu der folgenden Abbildung



Die Punkte liegen alle auf einer Linie. Man kann nun zwei Dinge tun

- eine Gerade einzeichnen, die diese Punkte verbindet und
- zu der Geraden eine Gleichung in der folgenden Form berechnen: y = a + b x



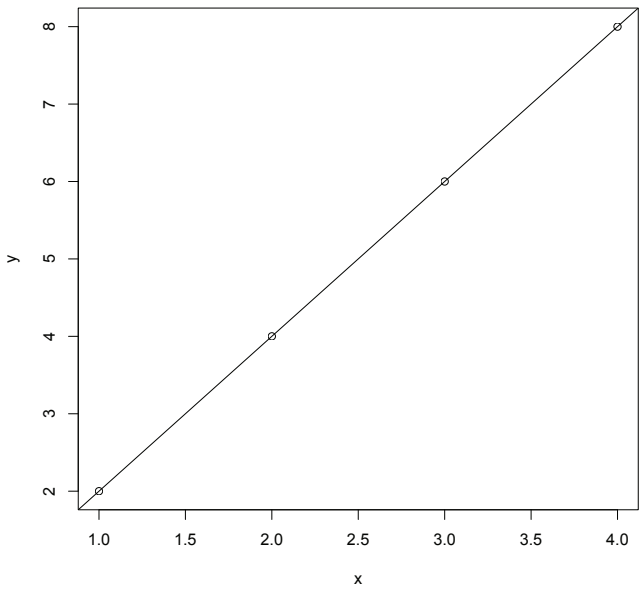
Punktewolken und Beziehungen

Früher, als man das Koordinatensystem von Hand gezeichnet hat, legte man ein Lineal auf die Punkte und verband diese. Mit einem Computer lässt man die Gerade mit R berechnen und mit der folgenden Anweisung zeichnen:

```
abline(lm(y ~ x))
```

Dabei werden mehrere Funktionen und Ausdrücke benutzt, sie werden unten detailliert erklärt und hier nur kurz vorgestellt:

- abline() dient zum Zeichnen einer Gerade,
- lm() berechnet eine Gerade, die durch eine Punktewolke geht,
- y ~ x legt fest, dass y von x abhängig sein soll.



Mit welcher Funktionsgleichung lässt sich nun die Gerade beschreiben? Auch das lässt sich mit wenig Aufwand berechnen, die Anweisung dazu lautet:

```
lm(y ~ x)
```

Sie liefert die folgenden Informationen:

```
Call:
lm(formula = y ~ x)

Coefficients:
(Intercept)           x
          0           2
```

Das bedeutet, die Funktionsgleichung hat die folgende Form: y = 2 x

Zur Kontrolle kann man für x irgendeine Zahl einsetzen, beispielsweise 20, es ergibt sich damit ein y-Wert von 40, auch dieser Wert wird auf der zuvor gezeichneten Geraden liegen, zusammen mit den anderen Punktepaaren. Gibt man für x den Wert 0 ein, dann erhält man den Schnittpunkt mit der y-Achse.

Wie gut ist repräsentiert diese Gerade nun die Wertepaare? Hierzu wird der sogenannte Korrelationskoeffizient R (ein anderer Begriff ist das Bestimmtheitsmaß) B (=R^2) berechnet. Das Bestimmtheitsmaß kann einen Wert zwischen 0 und 1 annehmen. Je näher er bei 1 liegt, umso besser werden die Punkte repräsentiert.

In diesem Beispiel kann man einen ziemlich hohen Wert erwarten, denn die Punkte liegen genau auf einer Geraden.

Die Anweisung zur Berechnung des Beistimmtheitsmaßes lautet:

```
summary(lm(x ~ y))
```

Sie liefert die folgenden Informationen, fett hervorgehoben ist das Bestimmtheitsmaß (R-squared): es hat den Wert 1. Dieser Wert ist so hoch, dass R zusätzlich eine Warnmeldung ausgibt (die letzten drei Zeilen).

```
Call:
lm(formula = x ~ y)

Residuals:
1 2 3 4
0 0 0 0

Coefficients:
             Estimate Std. Error t value Pr(>|t|)
(Intercept)      0.0        0.0      NA       NA
y                0.5        0.0     Inf   <2e-16 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 0 on 2 degrees of freedom
Multiple R-squared:     1,      Adjusted R-squared:     1
F-statistic:   Inf on 1 and 2 DF,  p-value: < 2.2e-16

Warnmeldung:
In summary.lm(lm(x ~ y)) :
  essentially perfect fit: summary may be unreliable
```

In vorangehenden einfachen Beispiel war der Zusammenhang der beiden Variablen funktional, so etwas wird in der Auswertung von Daten, die aus der Natur stammen, selten vorkommen, da es dort immer Streuung gibt, wie die folgenden Beispiele zeigen werden.

Lineare Regression: Daten von Kindern

In diesem Beispiel sollen Daten verwendet werden, die an Schulkindern gemessen wurden. Von zehn Kindern wurde Größe und Gewicht ermittelt.

groesse (cm)	135	145	139	142	137	137	134	144	135	146
gewicht (kg)	29,3	35,2	34,5	32,1	33,6	32,3	27,2	36,7	26,9	38,3

Die Daten stammen aus dem Buch „Kleine Enzyklopädie Mathematik“, Gellert, W. (Hrsg.), VEB Bibl. Inst. Leipzig, 13. Auflage (Beispieldaten auf S: 652)

Man wird vermuten, dass größere Kinder schwerer sind als kleinere Kinder, eine Analyse der Daten soll diese Vermutung entweder bestätigen oder sie verwerfen.

Ein erster Blick auf die Daten

Um die Daten für Größe und Gewicht in einem Koordinatensystem darstellen zu können, müssen sie zuerst in zwei Variablen gespeichert werden. Hierzu werden zwei neue Variablen angelegt und unter Verwendung der Funktion c() mit den Daten gefüllt (als Dezimaltrennzeichen muss der Punkt verwendet werden).

```
groesse <-c(135,145,139,142,137,137,134,144,135,146)

gewicht <-c(29.3,35.2,34.5,32.1,33.6,32.3,27.2,36.7,26.9,38.3)
```

Um die Inhalte der Variablen anzeigen zu lassen (z. B., um sie zu überprüfen), kann man die Namen der Variablen in R tippen und mit der Enter-Taste bestätigen:

```
> groesse
 [1] 135 145 139 142 137 137 134 144 135 146
> gewicht
 [1] 29.3 35.2 34.5 32.1 33.6 32.3 27.2 36.7 26.9 38.3
```

Als nächstes muss festgelegt werden, welche der Variablen die unabhängige (auf der x-Achse darzustellen) und welche die abhängige (auf der y-Achse darzustellen) sein soll.

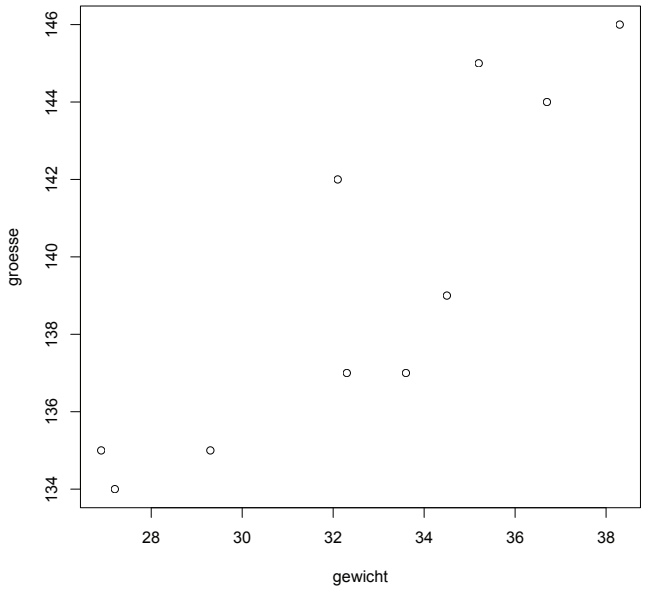
Da wir die Größe der Schulkinder aus dem Gewicht erklären möchten, soll das Gewicht unten auf der x-Achse dargestellt werden. Die Größe als abhängige Variable auf der y-Achse.

Punktewolke erzeugen

Mit der folgenden Anweisung erzeugt man das Koordinatensystem, das Gewicht wird auf der x-Achse, die Größe auf der y-Achse dargestellt:

```
plot(groesse ~ gewicht)
```

Die Tilde (~) steht zwischen der abhängigen Variable (groesse) und der unabhängigen Variablen (gewicht). Als Eselsbrücke kann die oben bereits vorgestellte allgemeine Geradengleichung dienen: y = a + b x, auch dort steht die abhängige Variable y zuerst.



Die eingangs formulierte Vermutung (mit zunehmendem Gewicht sind die Kinder auch größer) wird durch die Abbildung gestützt, die Punktewolke liegt schräg im Koordinatensystem von links unten nach rechts oben.

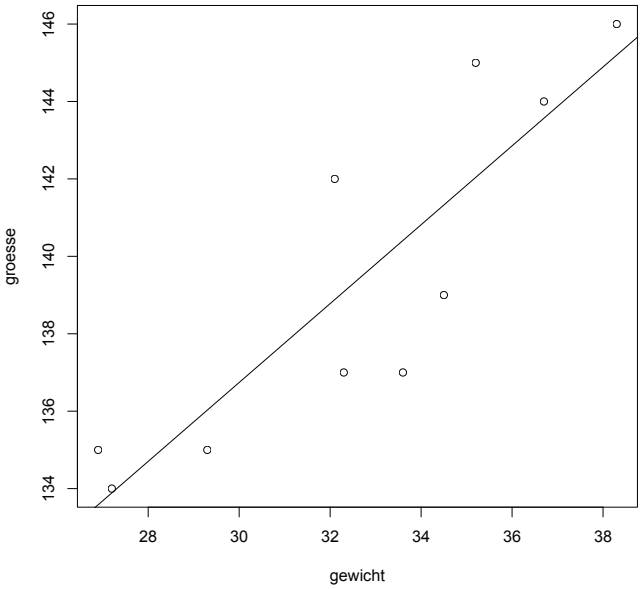
Ausgleichsgerade eintragen

Zu den Datenpunkten soll eine Ausgleichsgerade berechnet und eingetragen werden. Hierzu müssen wieder die Variablen angegeben und außerdem die zwei o. g. Funktionen von R verwendet werden:

- lm(): das ist die Abkürzung für ‚Lineares Modell'. Dieses einfache Modell wird mit der folgenden Funktion beschrieben: y = a + b x
- abline(): eine einfache Funktion zum Einfügen von Linien (von a nach b). Man kann abline() verwenden, um eine Gerade *irgendwo* in ein Bild einzufügen, mit der folgenden Anweisung wird beispielsweise eine vertikale rote Linie auf der Position x=30 eingefügt: abline(v=30, col= 'red')

Hier dagegen wird abline() kombiniert mit der Funktion lm() und den beiden Variablen aufgerufen: zunächst werden die Koordinaten der Ausgleichsgeraden unter Verwendung des linearen Modells berechnet (innere Klammer) und danach in die Punktewolke eingezeichnet (äußere Klammer).

```
abline(lm(groesse ~ gewicht))
```



Welche Werte für a und b (die beiden Koeffizienten aus der Gleichung) verwendet werden, lässt sich berechnen:

```
lm(groesse ~ gewicht)
```

ergibt (die Koeffizenten sind in der letzten Zeile)

```
Call:
lm(formula = groesse ~ gewicht)

Coefficients:
(Intercept)     gewicht
  106.177       1.019
```

Die Koeffizienten haben den Wert 106,2 und 1,02 (gerundet), somit ergibt sich die folgende Funktionsgleichung für die Ausgleichsgerade:

$$y = 106{,}2 + 1{,}02\ y$$

Damit ist die Berechnung der Regressionsgeraden abgeschlossen.

Zusammenfassende Statistik und B anzeigen lassen

Zu den Daten lassen sich zusammenfassende Informationen berechnen, zum Beispiel das Bestimmtheitsmaß (unten hervorgehoben). Die Anweisung

```
summary(lm(gewicht ~ groesse))
```

ergibt

```
Call:
lm(formula = gewicht ~ groesse)

Residuals:
    Min      1Q  Median      3Q     Max
-2.4495 -1.5359  0.3154  1.3019  2.7803

Coefficients:
             Estimate Std. Error t value Pr(>|t|)
(Intercept) -71.3745    20.6711  -3.453  0.00866 **
groesse       0.7459     0.1482   5.033  0.00101 **
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 2.002 on 8 degrees of freedom
```

```
Multiple R-squared:  0.76,    Adjusted R-squared:  0.73
F-statistic: 25.33 on 1 and 8 DF,  p-value: 0.001011
```

$R^2 = 0{,}73$
$R = 0{,}85$

Das Bestimmtheitsmaß ist mit 0,73 recht hoch. Somit lässt sich schlussfolgern, dass die Punktewolke durch die Ausgleichsgerade gut repräsentiert wird.

Nicht-lineare Regression: Holztrocknung

Nicht immer lässt sich eine Punktewolke sinnvollerweise durch eine Gerade repräsentieren, manchmal muss es eine Kurve sein.

Im folgenden Beispiel wurde der Feuchtegehalt von einem Stück Holz gemessen, es handelte sich um eine Scheibe von einem Haselnuss-Strauch mit einem Durchmesser von ca. 11 und einer Höhe von ca. 5 cm. Sie wurde nach dem Schnitt mit der Motorsäge in einen Raum mit ca. 20° C gebracht und die Holzfeuchte wurde über einen Zeitraum von 16 Tagen täglich gemessen, allerdings an drei Tagen nicht. Die Daten sehen folgendermaßen aus:

Datum	Gewicht
07.12.19	740
08.12.19	708
09.12.19	663
10.12.19	629
11.12.19	602
12.12.19	585
13.12.19	567
14.12.19	556
15.12.19	544
16.12.19	–
17.12.19	–
18.12.19	–
19.12.19	513
20.12.19	510
21.12.19	507
22.12.19	504

Die Daten wurden mit einer üblichen digitalen Küchenwaage gemessen, (auf 1 g genau). Die drei Striche symbolisieren die fehlenden Messwerte.

Das Gewicht fiel von 740 g auf 504 g, d. h. rund 1/3 des Gewichts bestand aus Wasser, das verdunstet ist.

Damit die Software R erfährt, dass es fehlende Werte gibt, verwendet man die Buchstabenfolge NA (für ‚not available'.

Da das Datum für die folgende Auswertung nicht gebraucht wird, werden fortlaufende Zahlen (1-16) für die Tage verwendet.

Mit der folgenden Anweisung werden die Daten in zwei Variablen geschrieben. Sie erhalten hier die Bezeichnung x und y, das ist nicht besonders originell, hilft aber bei der Zuordnung im Koordinatensystem:

Speichern der Feuchtewerte:

```
y<-c(740, 708, 663, 629, 602, 585, 567, 556, 544, NA, NA, NA, 513, 510, 507,
504)
```

Die fortlaufenden Zahlen von 1 bis 16 für die 16 Tage werden von R mit der folgenden Anweisung selbst erzeugt:

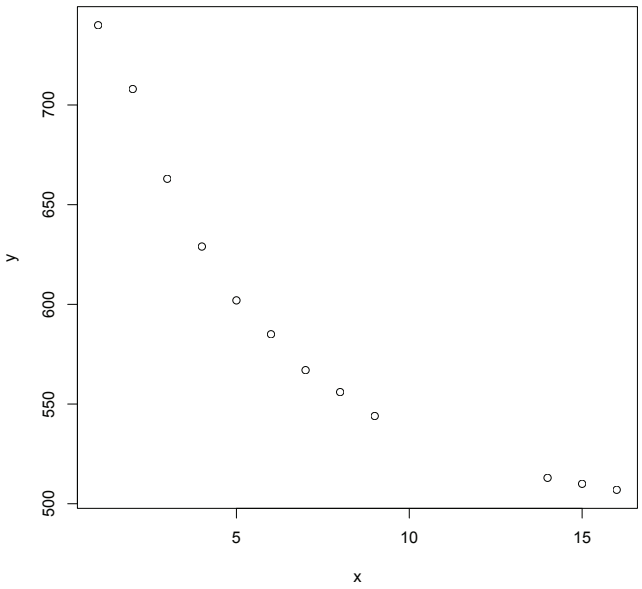
```
x<-c(1:16)
```

Alternativ hätte man schreiben können

```
x<-c(1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16)
```

Danach werden die Daten als Punktewolke angezeigt:

```
plot(y~x)
```



Es ist leicht zu erkennen, dass die Punkte in Form einer Kurve von links oben nach rechts unten verlaufen. Ein Ausgleich mit einer Geraden wäre zwar möglich, aber nicht sinnvoll.

Man erweitert die einfache Funktion zur Geradengleichung (y = a + b x) um einen quadratischen Teil, der einfach angehängt wird, oft ist es c x^2. Man kennt eine ähnliche Form ($y = x^2$) von der Parabelfunktion.

Nicht immer liefert x^2 die besten Ergebnissse. Hier wurde durch Ausprobieren mit 0,5 die beste Anpassung erreicht, so dass folgende Gleichung verwendet wurde:

$$y = a + b\,x + c\,x^{0{,}5}$$

Diese Funktionsgleichung muss nun für R in die folgende Anweisung ‚verpackt' werden:

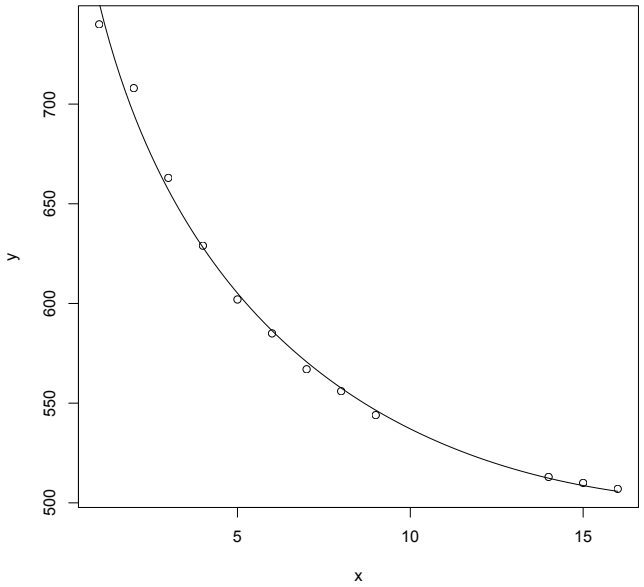
```
n <- nls(y ~ a + b * x + c * x^0.5, data = data.frame(x, y), start = c(a=1, b = 1,
c = 1))
```

Zur Erläuterung

- n ist eine neue Variable, in die Ergebnisse gespeichert werden
- nls() ist die Abkürzung für ‚non linear least squares', zu deutsch etwa: nicht linear, kleinste Quadrate (also die nichtlineare Funktion, bei der die quadrierten Abweichungen zwischen den Punkten und der Ausgleichsgeraden minimal sind)
- data = data.frame(x, y) legt fest, dass die Daten aus den Variablen x und y zu verwenden sind, sie werden zu einer Matrix zusammengefasst
- start = c(a=1, b = 1, c = 1) legt die Anfangswerte der drei Variablen a, b und c fest

Die Kurve selbst wird mit der Funktion curve() berechnet und eingezeichnet (dabei wird Bezug genommen auf die zuvor festgelegte Variable n).

```
curve(predict(n, newdata = data.frame(x = x)), add = TRUE)
```



Schließlich soll der Inhalt der Variablen n angezeigt werden.

```
summary(n)
```

liefert die folgenden Informationen (in der ersten Zeile die Funktionsgleichung, darunter die Werte für a, b und c):

```
Formula: y ~ a + b * x + c * x^0.5

Parameters:
  Estimate Std. Error t value Pr(>|t|)
a  914.349     13.816  66.181 2.07e-13 ***
b   20.482      2.163   9.468 5.63e-06 ***
c -184.070     11.405 -16.140 5.96e-08 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 6.323 on 9 degrees of freedom

Number of iterations to convergence: 1
Achieved convergence tolerance: 1.318e-06
  (4 observations deleted due to missingness)
```

Setzt man die Werte der errechneten Koeffizienten in die Gleichung ein, dann erhält man folgenden Ausdruck für die Funktionsgleichung:

$$y = 914 + 20x - 184x^{0,5}$$

Damit ist die Funktionsgleichung komplett.

Danach folgt ein Kasten zum Thema
Bestimmtheitsmaß und Residuen

Bestimmtheitsmaß und Residuen

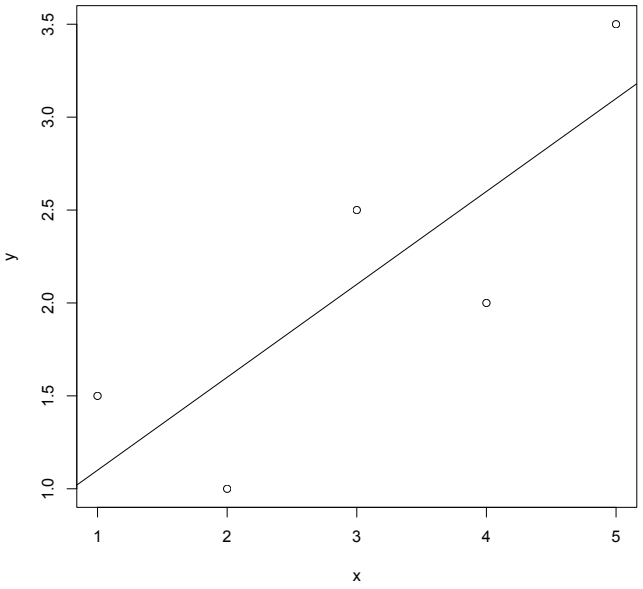
Das Bestimmtheitsmaß liefert wichtige Informationen darüber, wie gut eine Punktewolke durch eine Gerade oder Kurve repräsentiert wird.

Die Ausgleichsgerade wird durch die Methode der kleinsten Quadrate so berechnet, dass die Abstände der Datenpunkte zur Gerade minimal sind. Die folgenden zwei Beispiele zeigen das mit Punkten die einmal mehr und einmal weniger weit von der Ausgleichsgeraden entfernt sind:

a) Punkte liegen weit weg von der Ausgleichsgeraden

Variablen mit Daten füllen, Plot erzeugen, Gerade eintragen:

```
x<-c(1,2,3,4,5)
y<-c(1.5,1,2.5,2,3.5)
plot (y~x)
abline(lm(y ~ x))
```



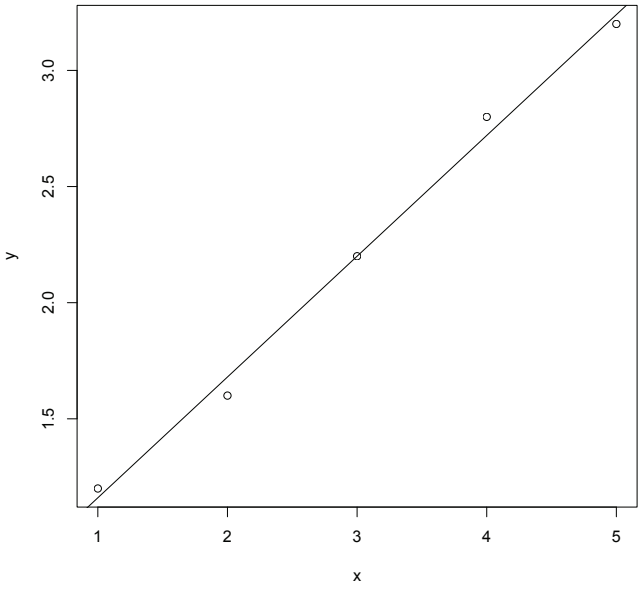

```
lm(y~x)
summary(lm(y ~ x))
Call:
lm(formula = y ~ x)

Residuals:
  1   2   3   4   5
0.4 -0.6  0.4 -0.6  0.4

Coefficients:
            Estimate Std. Error t value Pr(>|t|)
(Intercept)   0.6000     0.6633   0.905   0.4324
x             0.5000     0.2000   2.500   0.0877 .
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 0.6325 on 3 degrees of freedom
Multiple R-squared:  0.6757,   Adjusted R-squared:  0.5676
F-statistic:  6.25 on 1 and 3 DF,  p-value: 0.08771
```

Der Wert für R^2 ist mit 0,5676 recht niedrig.

b) Punkte liegen nahe an der Ausgleichsgeraden

Variablen mit Daten füllen, Plot erzeugen, Gerade eintragen:

```
x<-c(1,2,3,4,5)
y<-c(1.2, 1.6 ,2.2 ,2.8 ,3.2)
plot (y~x)

abline(lm(y ~ x))
```

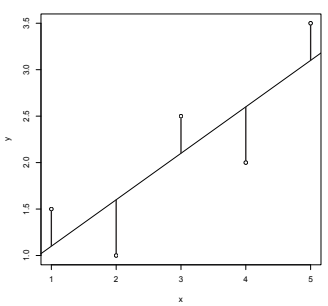



```
summary(lm(y ~ x))

Call:
lm(formula = y ~ x)

Residuals:
       1        2        3        4        5
 4.000e-02 -8.000e-02 -8.674e-17  8.000e-02 -4.000e-02

Coefficients:
            Estimate Std. Error t value Pr(>|t|)
(Intercept)  0.64000    0.07659   8.356 0.003594 **
x            0.52000    0.02309  22.517 0.000192 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 0.07303 on 3 degrees of freedom
Multiple R-squared:  0.9941,   Adjusted R-squared:  0.9922
F-statistic:   507 on 1 and 3 DF,  p-value: 0.0001918
```

Hier ist der Wert für R^2 mit 0,9922 sehr hoch!

Residuen

Man kann die Abstände der Punkte zu der Ausgleichsgerade sichtbar machen, indem man die Senkrechte eines jeden Punktes zur Geraden einzeichnet. Leider gibt es in R keine Funktion, die das im Handumdrehen macht, so soll eine manuell erstellte Zeichnung helfen:



Je kleiner die Residuen sind, umso besser repräsentiert die Ausgleichsgerade die Punktewolke.