

GESIS-Workshop
"Datenanalyse mit R"
DLR Berlin

Jan-Philipp Kolb

Freitag, 07. November, 2014



Inhaltsverzeichnis

Liebe auf den ersten Plot – Einfache Grafiken mit R

- Histogramm

- Barplots

- Boxplot

- Grafiken für bedingte, bi- und multivariate Verteilungen

Zusammenhangsmaße

- Zusammenhang zwischen stetigen Variablen

- Zusammenhang zwischen kategorialen Variablen

Die lineare Regression

Die logistische Regression

Ein zweiter Blick – Noch mehr Grafiken mit dem lattice Paket

Gliederung

Liebe auf den ersten Plot – Einfache Grafiken mit R

- Histogramm

- Barplots

- Boxplot

- Grafiken für bedingte, bi- und multivariate Verteilungen

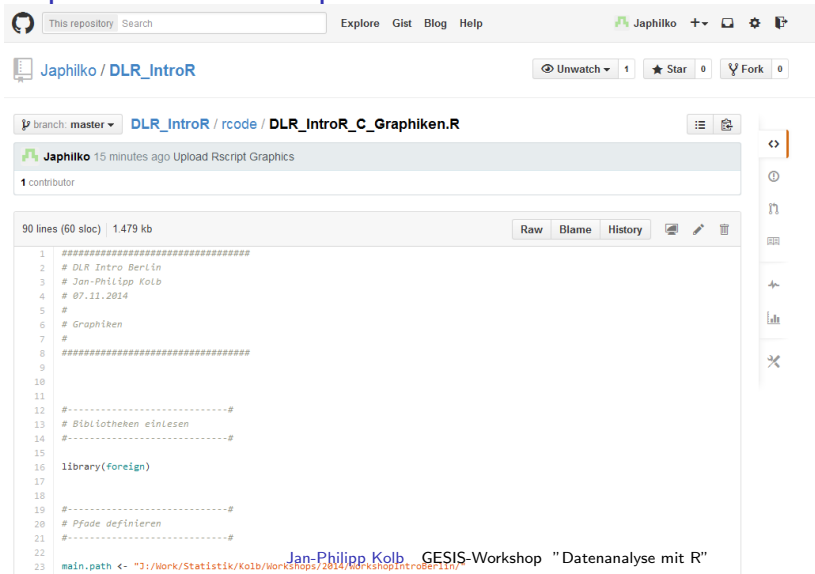
Zusammenhangsmaße

Die lineare Regression

Die logistische Regression

Ein zweiter Blick – Noch mehr Grafiken mit dem lattice Paket

R-Script - Einfache Graphiken



This repository Search Explore Gist Blog Help Japhilko + - ⌨ ⚙ 📄

Japhilko / DLR_IntroR Unwatch 1 Star 0 Fork 0

branch: master DLR_IntroR / rcode / DLR_IntroR_C_Graphiken.R

Japhilko 15 minutes ago Upload Rscript Graphics

1 contributor

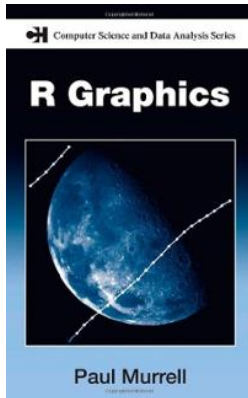
90 lines (60 sloc) 1.479 kb Raw Blame History

```
1 #####
2 # DLR Intro Berlin
3 # Jan-Philipp Kolb
4 # 07.11.2014
5 #
6 # Graphiken
7 #
8 #####
9
10
11
12 #-----#
13 # Bibliotheken einlesen
14 #-----#
15
16 library(foreign)
17
18
19 #-----#
20 # Pfade definieren
21 #-----#
22
23 main.path <- "J:/Work/Statistik/Kolb/Workshops/2014/workshopIntroBerlin/"
```

Ein Plot sagt mehr als 1000 Worte

- ▶ Grafisch gestützte Datenanalyse ist toll
- ▶ Gute Plots können zu einem besseren Verständnis beitragen
- ▶ Einen Plot zu generieren geht schnell
- ▶ Einen guten Plot zu machen kann sehr lange dauern
- ▶ Mit R Plots zu generieren macht Spaß
- ▶ Mit R erstellte Plots haben hohe Qualität
- ▶ Fast jeder Plottyp wird von R unterstützt
- ▶ R kennt eine große Menge an Exportformaten für Grafiken

Plot ist nicht gleich Plot



- ▶ Bereits das base Package bringt eine große Menge von Plot Funktionen mit
- ▶ Das lattice Package erweitert dessen Funktionalität
- ▶ Eine weit über diese Einführung hinausgehende Übersicht findet sich in Murrell, P (2006): R Graphics.

CRAN Task Views

- ▶ Zu einigen Themen sind alle Möglichkeiten in R zusammengestellt.
- ▶ Beispiel: Graphiken

CRAN Task View: Graphic Displays & Dynamic Graphics & Graphic Devices & Visualization

Maintainer: Nicholas Lewin-Koh

Contact: nikko at hailmail.net

Version: 2013-01-29

R is rich with facilities for creating and developing interesting graphics. Base R contains functionality for many plot types including coplots, mosaic plots, biplots, and the list goes on. There are devices such as postscript, png, jpeg and pdf for outputting graphics as well as device drivers for all platforms running R. [lattice](#) and [grid](#) are supplied with R's recommended packages and are included in every binary distribution. [lattice](#) is an R implementation of William Cleveland's trellis graphics, while [grid](#) defines a much more flexible graphics environment than the base R graphics.

Histogramm

- ▶ Die Funktion `hist()` plottet ein Histogramm der Daten
- ▶ Der Funktion muss mindestens ein Beobachtungsvektor übergeben werden
- ▶ `hist()` hat noch sehr viel mehr Argumente, die alle (sinnvolle) default values haben

Histogramm

Wir erstellen ein Histogramm der Variable `hheink`:

Die Funktion `hist()`

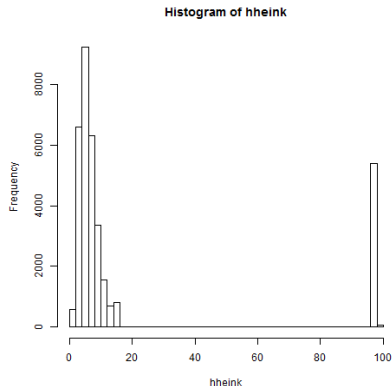
```
attach(midcar)

# Histogramm
?hist
hist(hheink)
```

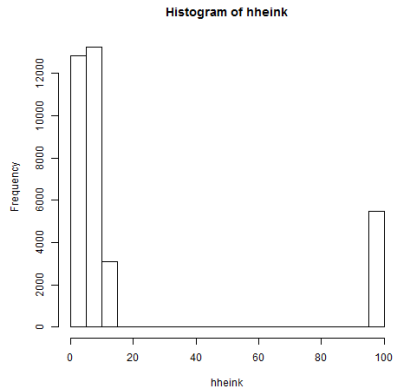
Histogramm

Die Zahl der Balken verändern:

```
hist(hheink, breaks=20)
```

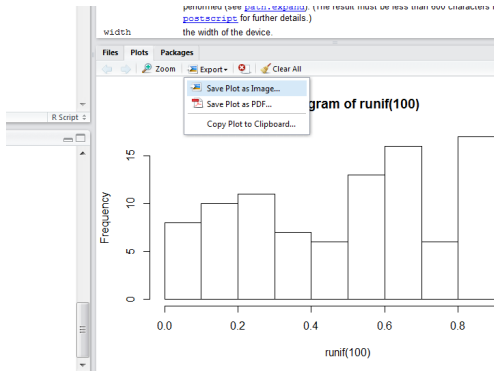


Histogramm



Graphik speichern

- ▶ Mit dem button Export kann man die Graphik speichern:



Alternative um Graphiken zu speichern

```
graph.path <- "C:/Graphiken/"  
setwd(graph.path)  
png("Graphikname.png")  
plot(x)  
dev.off()
```

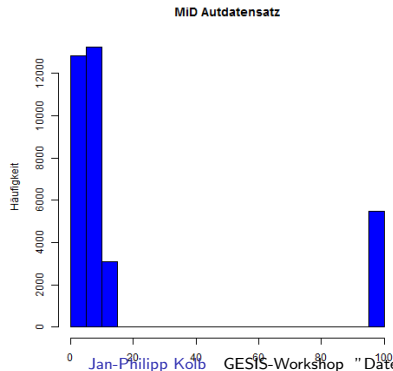
- ▶ Statt `png()` kann auch `pdf()` oder `jpeg()` verwendet werden.
- ▶ Diese Variante bietet sich an, wenn Graphik nach kleinen Veränderungen immer wieder abgespeichert werden muss.

Die wichtigsten Befehle bei einfachen plots

Argument	Bedeutung	Beispiel
main	Überschrift	main="Hallo Welt"
xlab	x-Achsenbeschriftung	xlab="x-Werte"
ylab	y-Achsenbeschriftung	ylab="y-Werte"
col	Farbe	col="blue"

Histogramm

```
hist(hheink, col="blue",  
     main="MiD Ausdatensatz",  
     ylab="Häufigkeit",  
     xlab="Haushaltseinkommen")
```



Barplot

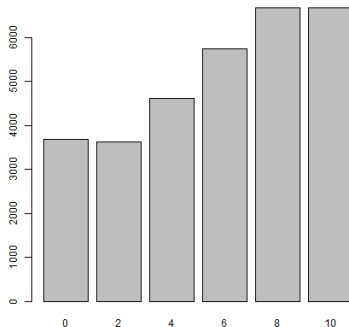
- ▶ Die Funktion `barplot()` erzeugt aus einer Häufigkeitstabelle einen Barplot
- ▶ Ist das übergebene Tabellen-Objekt zweidimensional wird ein bedingter Barplot erstellt

```
tabBland <- table(bland)
barplot(tabBland)
```


Barplots und barcharts

Mehr Farben!

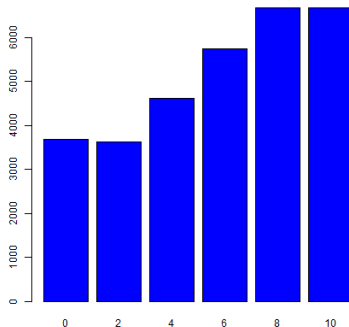
`barplot(tabScore)`



Barplots und barcharts

Mehr Farben!

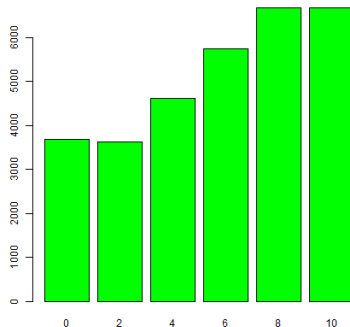
```
barplot(tabScore,col=rgb(0,0,1))
```



Barplots und barcharts

Mehr Farben!

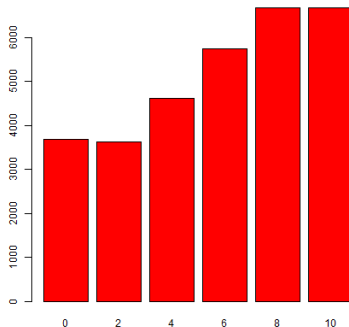
```
barplot(tabScore,col=rgb(0,1,0))
```



Barplots und barcharts

Mehr Farben!

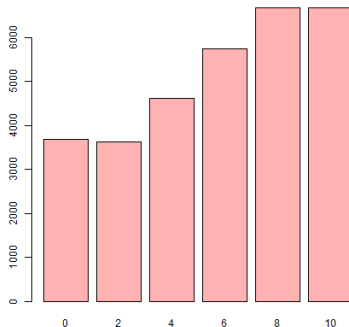
```
barplot(tabScore,col=rgb(1,0,0))
```



Barplots und barcharts

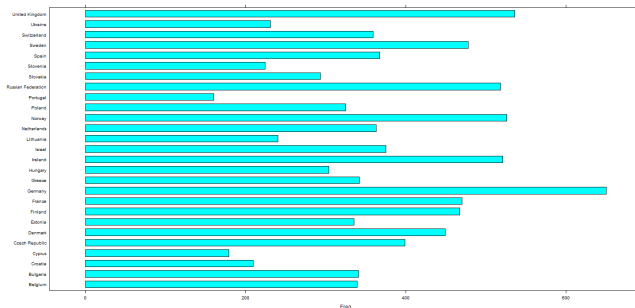
Mehr Farben!

```
barplot(tabScore,col=rgb(1,0,0,.3))
```



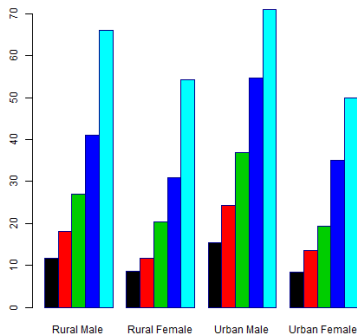
Barplots und barcharts

Eine erste lattice-Graphik `barchart(tabScore)`



Aufgabe 6- Barplot

- Laden Sie den Datensatz VADeaths und erzeugen Sie den folgenden plot:

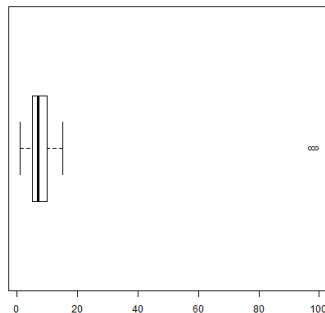


Boxplot

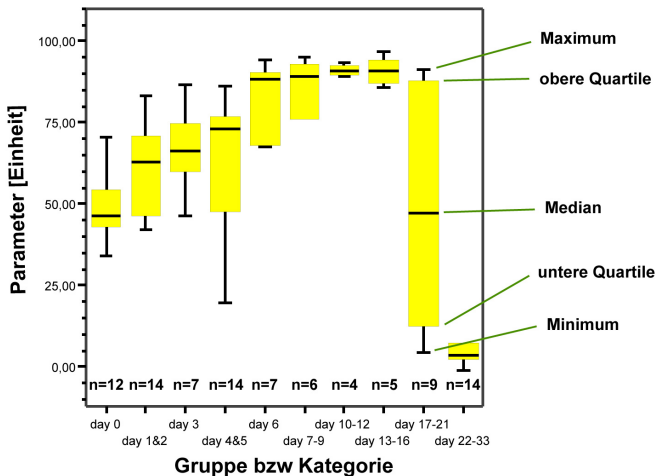
- ▶ Einen einfachen Boxplot erstellt man mit `boxplot()`
- ▶ Auch `boxplot()` muss mindestens ein Beobachtungsvektor übergeben werden

```
?boxplot
```

```
boxplot(hheink,  
horizontal=TRUE)
```



Boxplots



<http://edoc.hu-berlin.de/dissertationen/gruenwald-andreas-2005-01-17/HTML/chapter2.html>

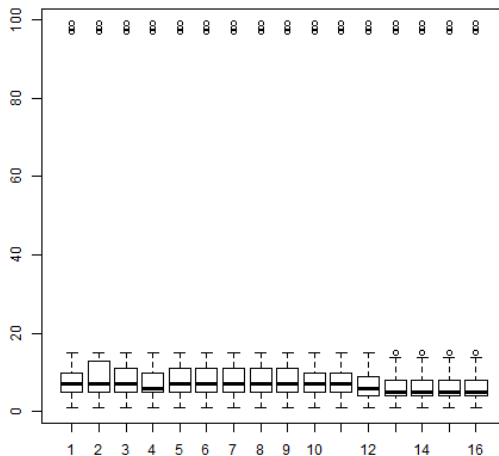
Gruppierte Boxplots

- ▶ Ein sehr einfacher Weg, einen ersten Eindruck über bedingte Verteilungen zu bekommen ist über sog. Gruppierte notched Boxplots
- ▶ Dazu muss der Funktion `boxplot()` ein sog. Formel-Objekt übergeben werden
- ▶ Die bedingende Variable steht dabei auf der rechten Seite einer Tilde

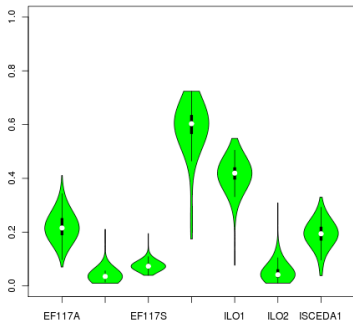
Die Funktion `boxplot()`

```
boxplot(hheink~bland)
```

Gruppierte Boxplots



Violinplot - *library(vioplot)*



- ▶ Baut auf Boxplot auf
- ▶ Zusätzlich Informationen über Dichte der Daten
- ▶ Dichte wird über Kernel Methode berechnet.
- ▶ weißer Punkt - Median
- ▶ Je weiter die Ausdehnung, desto größer ist die Dichte an dieser Stelle.

Scatterplots

- ▶ Ein einfacher two-way scatterplot kann mit der Funktion `plot()` erstellt werden
- ▶ `plot()` muss mindestens ein `x` und ein `y` Beobachtungsvektor übergeben werden
- ▶ Um die Farbe der Plot-Symbole anzupassen gibt es die Option `col` (Farbe als character oder numerisch)
- ▶ Die Plot-Symbole selbst können mit `pch` (plotting character) angepasst werden (character oder numerisch)
- ▶ Die Achsenbeschriftungen (labels) werden mit `xlab` und `ylab` definiert

Datensatz OECD

Datensatz enthält folgende Variablen (Stand 2009), die das Wohlergehen von Kindern in Mitgliedstaaten messen.

Einkommen ∅ Einkommen der Eltern [in tsd USD pro Kind]

Armut Anteil [immer in %] an Kindern in armen Elternhaus

Bildung Anteil Kinder, ohne Grundausrüstung (Bücher, Schreibtisch, Computer, Internet) für Bildung

WenigRaum Anteil an Kindern, die auf zu wenig Raum wohnen

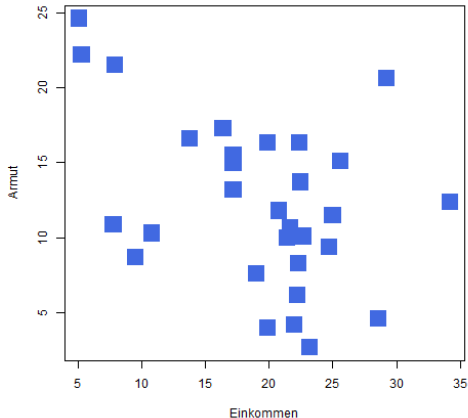
Alkohol Anteil an 13-15 jährigen Jugendlichen, die mindestens zweimal betrunken waren

<http://www.uni-leipzig.de/~zuber/teaching/ws12/r-kurs/praxis/oecdM.csv>

- └ Liebe auf den ersten Plot – Einfache Grafiken mit R
- └ Grafiken für bedingte, bi- und multivariate Verteilungen

Scatterplot

```
plot(oecd$Einkommen, oecd$Armut, xlab="Einkommen",  
     ylab="Armut", pch=15, col="royalblue")
```



Aufgabe 7 - Datenanalyse

- ▶ Lesen Sie den Datensatz `oecd` mit folgender Funktion ein:

```
data<-read.csv(file="http://www.uni-leipzig.de/~zuber/  
teaching/ws12/r-kurs/praxis/oecdM.csv", header=TRUE)
```

- ▶ Überprüfen Sie die Dimension der OECD-Daten.
- ▶ Berechnen Sie die Mittelwerte und Varianzen der einzelnen Variablen mit einem geeigneten `apply` Befehl.
- ▶ In welchem Land waren die meisten Jugendlichen mindestens zweimal betrunken? Wie hoch ist der maximale Prozentsatz?
- ▶ In welchem Land ist die Sterblichkeit am geringsten? Wie hoch ist sie in diesem Land?
- ▶ Erstellen Sie einen neuen Datensatz, der aufsteigend nach dem Einkommen geordnet ist. Speichern Sie diesen in einer neuen `.csv` Datei

Gliederung

Liebe auf den ersten Plot – Einfache Grafiken mit R

Zusammenhangsmaße

Zusammenhang zwischen stetigen Variablen

Zusammenhang zwischen kategorialen Variablen

Die lineare Regression

Die logistische Regression

Ein zweiter Blick – Noch mehr Grafiken mit dem lattice Paket

Edgar Anderson's Iris Daten

petal length and width	Blütenblatt Länge und Breite
sepal length and width	Kelchblatt Länge und Breite

```
> head(iris)
```

	sepal.Length	sepal.width	petal.Length	petal.width	species
1	5.1	3.5	1.4	0.2	setosa
2	4.9	3.0	1.4	0.2	setosa
3	4.7	3.2	1.3	0.2	setosa
4	4.6	3.1	1.5	0.2	setosa
5	5.0	3.6	1.4	0.2	setosa
6	5.4	3.9	1.7	0.4	setosa

Edgar Anderson's Iris Daten



WIKIPEDIA
The Free Encyclopedia

[Main page](#)

[Contents](#)

[Featured content](#)

[Current events](#)

[Random article](#)

[Donate to Wikipedia](#)

[Wikimedia Shop](#)

▼ [Interaction](#)

[Help](#)

[About Wikipedia](#)

[Community portal](#)

[Recent changes](#)

[Contact page](#)

Article [Talk](#)

[Read](#)

[Edit](#)

[View history](#)

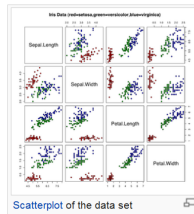
[Create account](#) [Log in](#)

Iris flower data set

From Wikipedia, the free encyclopedia

The **Iris flower data set** or **Fisher's Iris data set** is a [multivariate data set](#) introduced by [Sir Ronald Fisher](#) (1936) as an example of [discriminant analysis](#).^[1] It is sometimes called **Anderson's Iris data set** because [Edgar Anderson](#) collected the data to quantify the morphologic variation of [Iris](#) flowers of three related species.^[2] Two of the three species were collected in the [Gaspé Peninsula](#) "all from the same pasture, and picked on the same day and measured at the same time by the same person with the same apparatus".^[3]

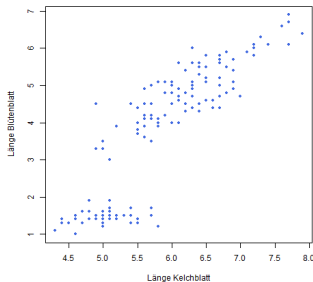
The data set consists of 50 samples from each of three species of *Iris* (*Iris setosa*, *Iris virginica* and *Iris versicolor*). Four [features](#) were measured from each sample: the length and the width of the [sepals](#) and [petals](#), in centimetres. Based on the combination of these four features, Fisher developed a [linear discriminant model](#) to distinguish the species from each other.



Pearson Korrelationskoeffizient

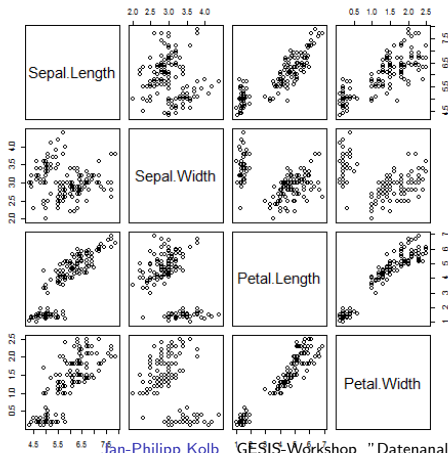
```
cor(iris$Sepal.Length, iris$Petal.Length)
```

- ▶ Korrelation zwischen Länge Kelchblatt und Blütenblatt 0,87
- ▶ Der Pearson'sche Korrelationskoeffizient ist die default methode in `cor()`:



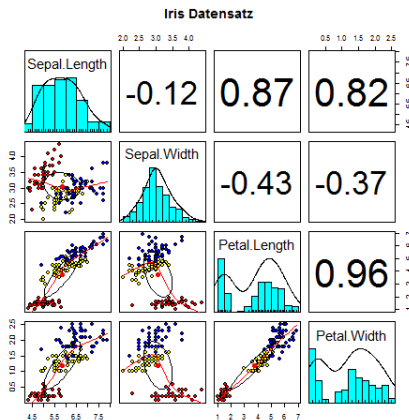
Zusammenhang zwischen mehreren Variablen

```
pairs(iris[,1:4])
```



Zusammenhang zwischen mehreren Variablen

```
pairs.panels(iris[1:4], bg=c("red", "yellow", "blue")  
[iris$Species], pch=21, main="Iris Datensatz")
```



Verschiedene Korrelationskoeffizienten

Pearson Korrelationskoeffizient

```
cor(iris[,1:4])
```

Kendall's *tau* (Rangkorrelation)

```
cor(iris[,1:4], method = "kendall")
```

Spearman's ρ (Rangkorrelation)

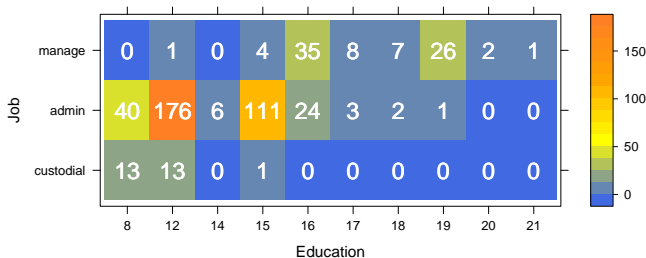
```
cor(iris[,1:4], method = "spearman")
```

Zusammenhang zwischen kategorialen Variablen

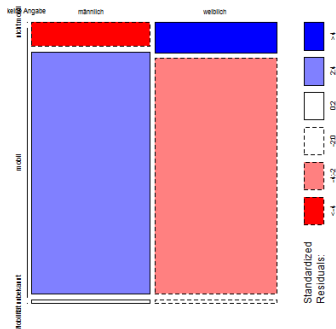
- ▶ `chisq.test()` testet, ob zwei kategoriale Merkmale stochastisch unabhängig sind.
- ▶ Getestet wird gegen die Nullhypothese der Gleichverteilung

Levelplot Datensatz BankWages

```
levelplot(table(BankWages$education, BankWages$job))
```



Visualisierung von Zusammenhängen zwischen kategorialen Variablen



- ▶ Flächen werden entsprechend der Residuen eingefärbt.
- ▶ `mosaicplot()`
- ▶ Pearson Residuen:

$$r_{ij} = \frac{n_{ij} - \hat{n}_{ij}}{\sqrt{\hat{n}_{ij}}}$$

Gliederung

Liebe auf den ersten Plot – Einfache Grafiken mit R

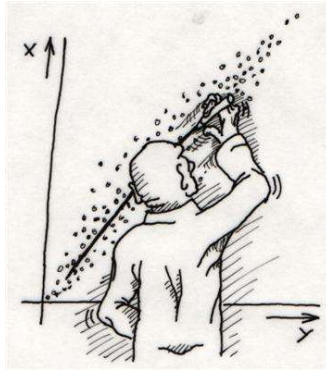
Zusammenhangsmaße

Die lineare Regression


Die logistische Regression

Ein zweiter Blick – Noch mehr Grafiken mit dem lattice Paket


Die lineare Regression





R Script - lineare Regression


 This repository Search


Explore Gist Blog Help

 Japhilko + ▾








 Japhilko / DLR_IntroR

Unwatch ▾ 1

★ Star 0

Fork 0

branch: master ▾ DLR_IntroR / rcode / DLR_IntroR_E_lineareRegression.R

 Japhilko 22 seconds ago Upload Rcode

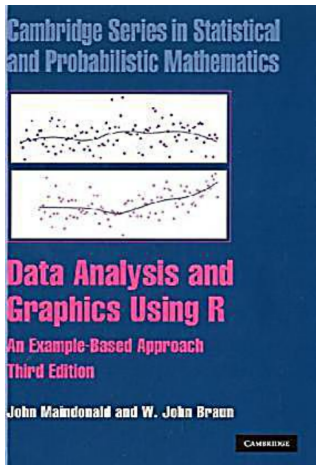
1 contributor

108 lines (69 slocl) | 2.448 kb

Raw Blame History

```
1 #####
2 # DLR Intro Berlin
3 # Jan-Philipp Kolb
4 # 07.11.2014
5 #
6 # Lineare Regression
7 #
8 #####
9
10 #-----#
11 # Pfade angeben
12 #-----#
13
14 main.path <- "J:/Work/Statistik/Kolb/Workshops/2014/WorkshopIntroBerlin/"
15
16 graph.path <- paste(main.path,"Rfolien/figure/",sep="")
17
18 #-----#
19 # Bibliotheken
20 #-----#
21
22 library(lattice)
```

Literatur Regression



1. Einführung in R
2. Datenanalyse
3. Statistische Modelle
4. Inferenzkonzepte
5. Regression mit einem Prädiktor
6. Multiple lineare Regression
7. Ausweitung des linearen Modells
8. ...

Lineare Regression in R - Beispieldatensatz

Lawn Roller Data

The `roller` data frame has 10 rows and 2 columns. Different weights of roller were rolled over different parts of a lawn, and the depression was recorded.

Description

The `roller` data frame has 10 rows and 2 columns. Different weights of roller were rolled over different parts of a lawn, and the depression was recorded.

Usage

`roller`

The data frame contains the following columns:

Format

This data frame contains the following columns:

`weight`

`depth` a numeric vector consisting of the roller weights

`depression`

the depth of the depression made in the grass under the roller

```
library(DAAG)
data(roller)
?roller
```


Das lineare Regressionsmodell in R

Schätzen eines Regressionsmodells:

```
roller.lm <- lm(depression ~ weight, data = roller)
```

So bekommt man die Schätzwerte:

```
summary(roller.lm)
```

Falls das Modell ohne Intercept geschätzt werden soll:

```
lm(depression ~ -1 + weight, data = roller)
```

Summary des Modells

```
summary(roller.lm)
```

Call:

```
lm(formula = depression ~ weight, data = roller)
```

Residuals:

Min	1Q	Median	3Q	Max
-8.180	-5.580	-1.346	5.920	8.020

Coefficients:

	Estimate	Std. Error	t value	Pr(> t)
(Intercept)	-2.0871	4.7543	-0.439	0.67227
weight	2.6667	0.7002	3.808	0.00518 **

signif. codes: 0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 6.735 on 8 degrees of freedom

Multiple R-squared: 0.6445, Adjusted R-squared: 0.6001

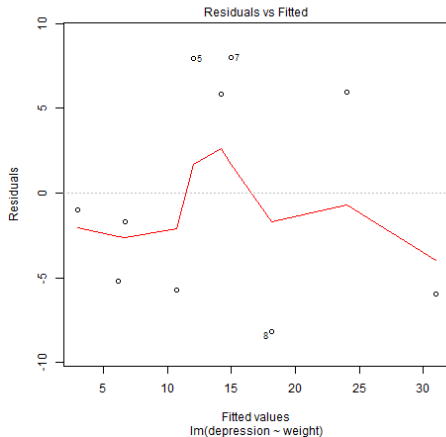
F-statistic: 14.5 on 1 and 8 DF, p-value: 0.005175

R arbeitet mit Objekten

- ▶ `roller.lm` ist nun ein spezielles Regressions-Objekt
- ▶ Auf dieses Objekt können nun verschiedene Funktionen angewendet werden

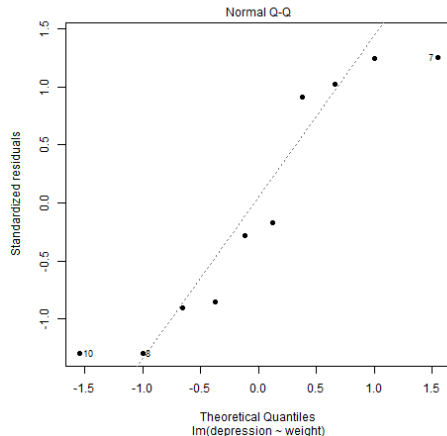
```
predict(roller.lm) # Vorhersage  
resid(roller.lm)  # Residuen
```

Residuenplot



- ▶ Sind Annahmen des linearen Regressionsmodells verletzt?
- ▶ Dies ist der Fall, wenn ein Muster abweichend von einer Linie zu erkennen ist.
- ▶ Hier ist der Datensatz sehr klein

Residuenplot



- ▶ Wenn die Residuen normalverteilt sind sollten sie auf einer Linie liegen.

Linkliste - lineare Regression

- ▶ Auf dem Kurs an der Uni Leipzig von Verena Zuber basieren auch viele der Aufgaben in diesem Workshop:
<http://www.uni-leipzig.de/~zuber/teaching/ws09/r-kurs/theorie/Kurs9.pdf>
- ▶ Eine der vielen interessanten Blogs auf r-bloggers:
<http://www.r-bloggers.com/r-tutorial-series-simple-linear-regression/>
- ▶ Komplettes Buch von Faraway (sehr intuitiv geschrieben):
<http://cran.r-project.org/doc/contrib/Faraway-PRA.pdf>
- ▶ Gute Einführung auf Quick-R:
<http://www.statmethods.net/stats/regression.html>

Aufgabe 8 - lineare Regression

Datensatz toycars - Paket DAAG

Beschrieben wird Wegstrecke, dreier Spielzeugautos die in unterschiedlichen Winkeln Rampe herunterfahren.

- ▶ `angle`: Winkel der Rampe
- ▶ `distance`: Zurückgelegte Strecke des Spielzeugautos
- ▶ `car`: Autotyp (1, 2 oder 3)

Quelle: <http://www.uni-leipzig.de/~zuber/teaching/ws09/r-kurs/praxis/U9.pdf>

Aufgabe 8 - lineare Regression

- (a) Installieren und laden Sie das Paket **DAAG**.
- (b) Speichern Sie den Datensatz "*toycars*" in einem dataframe **data** ab und wandeln Sie die Variable "**car**" des Datensatzes in einen Faktor (**as.factor**) um.
- (c) Erstellen Sie drei Boxplots, die die zurückgelegte Strecke getrennt nach dem Faktor "**car**" darstellen.
- (d) Schätzen Sie für **jedes** der 3 Autos **separat** die Parameter des folgenden linearen Modells mit Hilfe der Funktion "**lm()**"

$$\text{distance}_i = \beta_0 + \beta_1 \cdot \text{angle}_i + \varepsilon_i$$

- (e) Überprüfen Sie deskriptiv den Fit der drei Modelle, indem Sie die Regressiongerade in einen Plot von *distance* gegen *angle* einfügen. Deutet das R^2 jeweils auf eine gute Modellanpassung hin?
- (f) Führen Sie weitere deskriptive Diagnosen mit Hilfe der **plot.lm()** Funktion durch. Besteht ein linearer Zusammenhang? Sind die Residuen normalverteilt? Haben die Fehler gleiche Varianz?

Quelle: <http://www.uni-leipzig.de/~zuber/teaching/ws09/r-kurs/praxis/U9.pdf>

Gliederung

Liebe auf den ersten Plot – Einfache Grafiken mit R

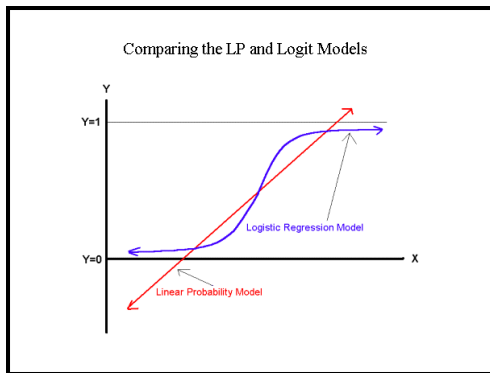
Zusammenhangsmaße

Die lineare Regression

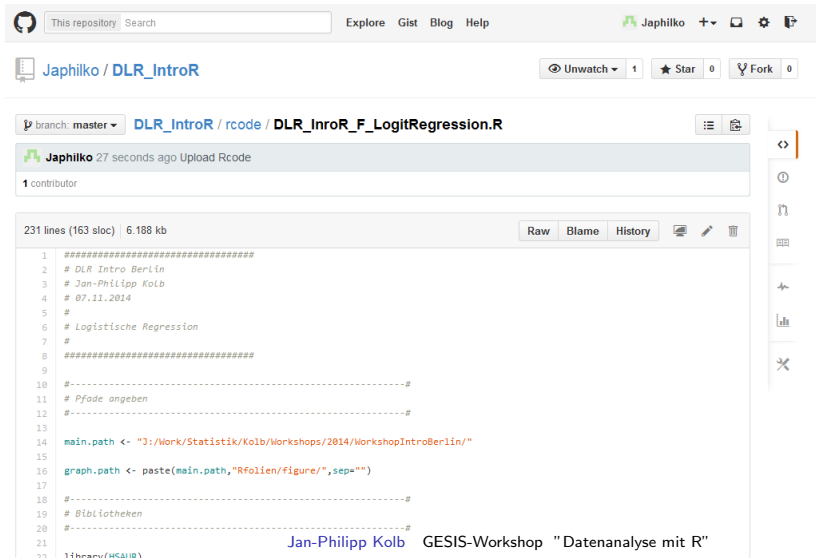
Die logistische Regression

Ein zweiter Blick – Noch mehr Grafiken mit dem lattice Paket

Die logistische Regression



R Skript logistische Regression



This repository Search Explore Gist Blog Help Japhilko + - ⚙️

Japhilko / DLR_IntroR Unwatch 1 Star 0 Fork 0

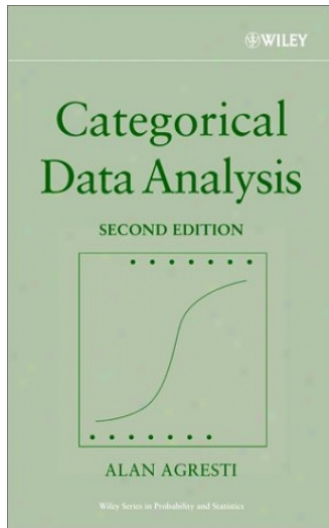
branch: master DLR_IntroR / rcode / DLR_IntroR_F_LogitRegression.R

Japhilko 27 seconds ago Upload Rcode 1 contributor

231 lines (163 sloc) 6.188 kb Raw Blame History

```
1 #####
2 # DLR Intro Berlin
3 # Jan-Philipp Kolb
4 # 07.11.2014
5 #
6 # Logistische Regression
7 #
8 #####
9
10 #-----#
11 # Pfade angeben
12 #-----#
13
14 main.path <- "J:/Work/Statistik/Kolb/Workshops/2014/WorkshopIntroBerlin/"
15
16 graph.path <- paste(main.path,"Rfolien/figure/",sep="")
17
18 #-----#
19 # Bibliotheken
20 #-----#
21
22 library(HSAUR)
```

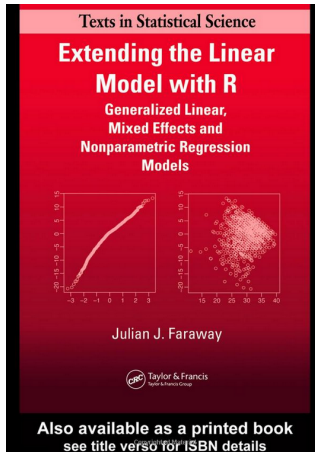
Literatur zum Thema



- ▶ Sehr intuitiv geschriebenes Buch
- ▶ Sehr ausführliches begleitendes Skript von Thompson
- ▶ Das Skript eignet sich um die kategoriale Datenanalyse nachzuvollziehen

<https://home.comcast.net/~lthompson221/Splusdiscrete2.pdf>

Literatur zu logistischer Regression in R



home.comcast.net/~lthompson221/Spplusdiscrete2.pdf

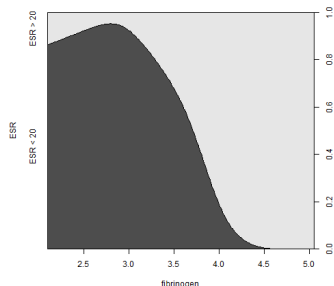
- ▶ Logistische Regressionen gut erklärt
- ▶ Beispiele mit R-code

Binäre AVs mit `glm()`

- ▶ Die logistische Regression gehört zur Klasse der generalisierten linearen Modelle (GLM)
- ▶ Die Funktion zur Schätzung eines Modells dieser Klasse in heißt `glm()`
- ▶ `glm()` muss 1. ein Formel-Objekt mitgegeben werden und 2. die Klasse (binomial, gaussian, Gamma) samt link-Funktion (logit, probit, cauchit, log, cloglog)

Logistische Regression mit R

```
data("plasma", package = "HSAUR")  
  
cdplot(ESR ~ fibrinogen, data = plasma)  
  
plasma_glm_1 <- glm(ESR ~ fibrinogen, data = plasma,  
                    family = binomial())
```



Generalisierte Regression mit R - weitere Funktionen

Logistisches Modell mit Probit-Link:

```
probitmod <- glm(cbind(damage,6-damage) ~ temp,  
family=binomial(link=probit), orings)
```

Regression mit Zähldaten:

```
modp <- glm(Species ~ .,family=poisson,gala)
```

Proportional odds logistic regression im Paket `library(MASS)`:

```
house.plr<-polr(Sat~Infl,weights=Freq,data=housing)
```


Linkliste - logistische Regression

- ▶ Einführung in logistische Regression:
`http://ww2.coastal.edu/kingw/statistics/
R-tutorials/logistic.html`
- ▶ Code zum Buch von Faraway:
`http://www.maths.bath.ac.uk/~jjf23/ELM/scripts/
binary.R`

Gliederung

Liebe auf den ersten Plot – Einfache Grafiken mit R

Zusammenhangsmaße

Die lineare Regression

Die logistische Regression

Ein zweiter Blick – Noch mehr Grafiken mit dem lattice Paket

R Script - Lattice Graphiken

The screenshot shows a GitHub repository page for the user 'Japhilko' and the repository 'DLR_IntroR'. The file 'DLR_IntroR_G_Lattice.R' is selected, showing its content. The file is 99 lines long and 2.612 kb in size. The content is an R script with comments in German and R code. The script includes a header with the title 'Einführung in R' and the author 'Jan-Philipp Kolb'. It also includes a section for 'General Information' and a section for 'Pfade angeben' (Paths specified). The script defines variables for 'scriptname' and 'author', and a path for 'graph.path'.

branch: master ▾ **DLR_IntroR** / **rcode** / **DLR_IntroR_G_Lattice.R**

Japhilko 2 minutes ago Upload Rcode

1 contributor

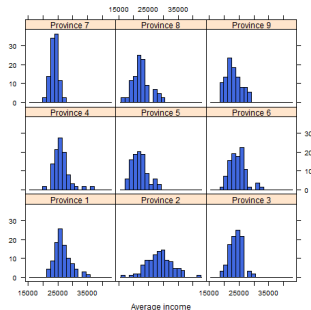
99 lines (63 sloc) | 2.612 kb

Raw Blame History

```
1  #-----#
2  # Einführung in R
3  # Jan-Philipp Kolb
4  #
5  # Lattice Graphics
6  #
7  # 29.04.2014
8  #-----#
9
10
11 #-----#
12 # General Information
13 #-----#
14
15 scriptname <- "IntroR_E_latticeGraphics.R"
16 author <- "Jan-Philipp Kolb"
17
18 #-----#
19 # Pfade angeben
20 #-----#
21
22 graph.path <- "J:/Work/Statistik/Kolb/Workshops/2014/WorkshopEinfuehrungR/Folien/figure"
```

Das lattice-Paket

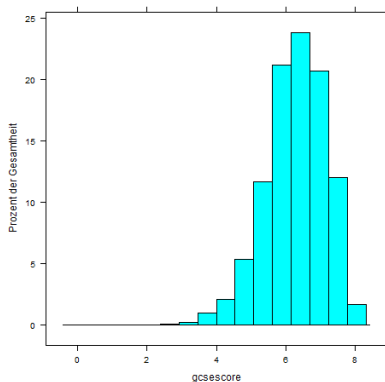
- It is designed to meet most typical graphics needs with minimal tuning, but can also be easily extended to handle most nonstandard requirements.



<http://stat.ethz.ch/R-manual/R-devel/library/lattice/html/Lattice.html>

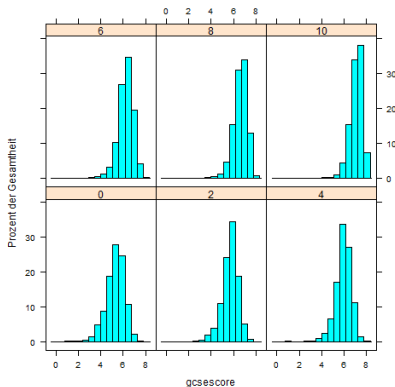
Histogramm mit Lattice

```
histogram(~ gcsescore, data = Chem97)
```



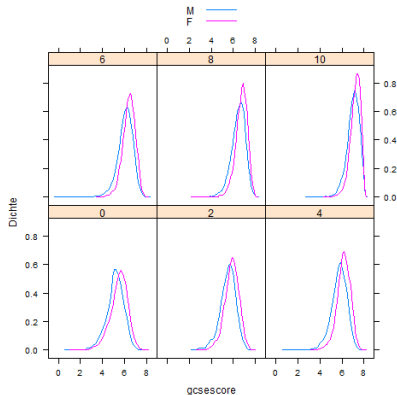
Histogramm mit Lattice

```
histogram(~ gcsescore | factor(score), data = Chem97)
```



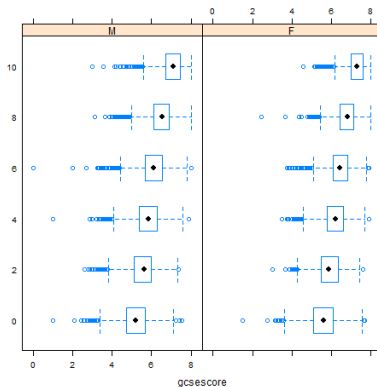
Die Dichte mit Lattice zeichnen

```
densityplot(~ gcsescore | factor(score), Chem97,  
groups=gender, plot.points=FALSE, auto.key=TRUE)
```



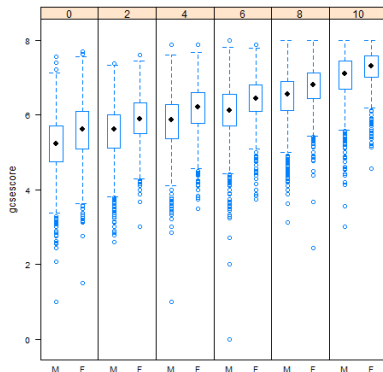
Boxplot mit Lattice zeichnen

```
bwplot(factor(score) ~ gcsescore | gender, Chem97)
```



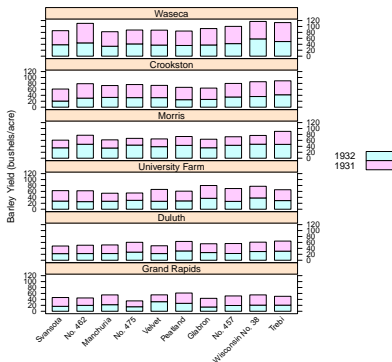
Boxplot mit Lattice zeichnen

```
bwplot(gcsescore ~ gender | factor(score), Chem97,  
       layout = c(6, 1))
```

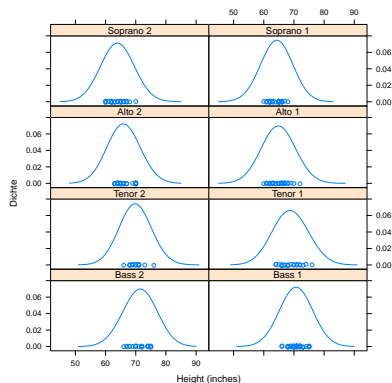


Univariate Plots

barchart()

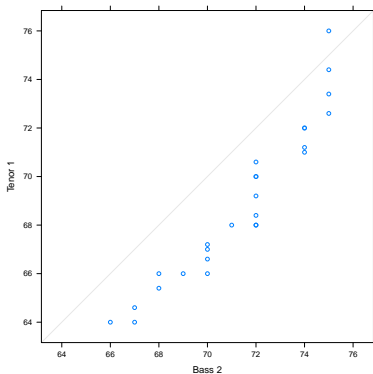


densityplot()

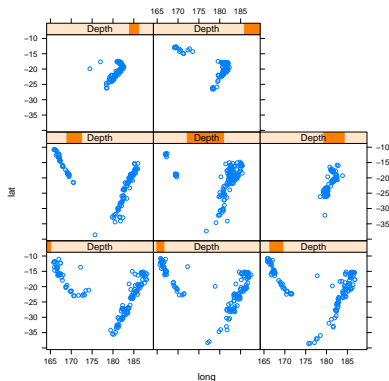


Bivariate Plots

qqplot()

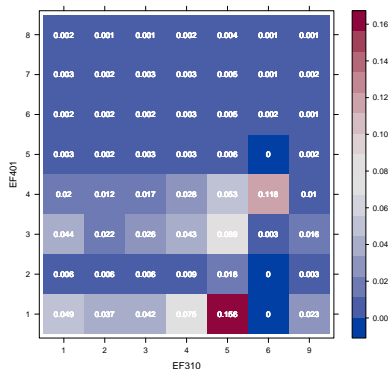


xyplot()

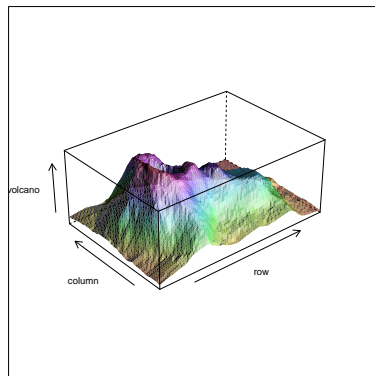


Trivariate Plots

`levelplot()`

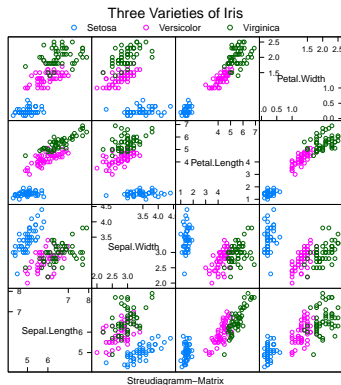


`wireframe()`

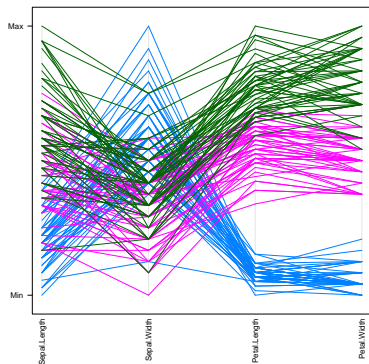


Hypervariate Plots

`splom()`



`parallel()`

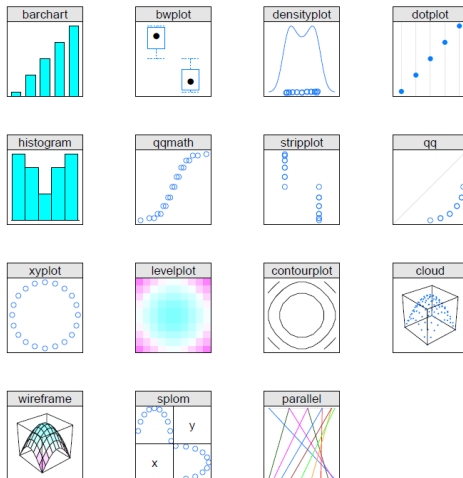


LatticeBefehle

Function	Default Display
histogram()	Histogram
densityplot()	Kernel Density Plot
qqmath()	Theoretical Quantile Plot
qq()	Two-sample Quantile Plot
stripplot()	Stripchart (Comparative 1-D Scatterplots)
bwplot()	Comparative Box-and-Whisker Plots
dotplot()	Cleveland Dot Plot
barchart()	Bar Plot
xyplot()	Scatterplot
splom()	Scatterplot Matrix
contourplot()	Contour Plot of Surfaces
levelplot()	False Color Level Plot of Surfaces
wireframe()	Three-dimensional Perspective Plot of Surfaces
cloud()	Three-dimensional Scatterplot
parallel()	Parallel Coordinates Plot

Quelle: http://www.isid.ac.in/~deepayan/R-tutorials/labs/04_lattice_lab.pdf

LatticeBefehle



Quelle: Universität Trier (2013) Statistical Programming with R

Wichtige Bibliotheken für die graphische Datenanalyse

Bibliothek	Thema
lattice	Lattice is a powerful and elegant high-level data visualization system
vcd	Visualization techniques, data sets, summary and inference procedures aimed particularly at categorical data.
ggplot2	An implementation of the grammar of graphics in R.

Aufgabe 9 - Datenanalyse II

- ▶ Laden Sie einen Datensatz Ihrer Wahl - entweder einen eigenen oder einen der vorgestellten Datensätze
- ▶ Berechnen Sie einfache Statistiken auf den wichtigsten Variablen (Mittelwert, Median, Standardabweichung)
- ▶ Erzeugen Sie eine zweidimensionale Häufigkeitstabelle
- ▶ Führen Sie eine Regression auf den Daten durch
- ▶ Erzeugen Sie einen Lattice-plot

Vielen Dank für die Aufmerksamkeit

