

Statistische Software (R)

Paul Fink, M.Sc., Eva Endres, M.Sc.

Institut für Statistik

Ludwig-Maximilians-Universität München

Verteilungen und Zufallszahlen



Übersicht Statistik-Funktionen

Funktion	Beschreibung
<code>mean()</code>	arithmetische Mittel
<code>median()</code>	Median
<code>exp(mean(log()))</code>	Geometrisches Mittel
<code>quantile()</code>	empirische Quantile
<code>var()</code>	Stichproben-Varianz
<code>sd()</code>	Stichproben-Standardabweichung
<code>range()</code>	Minimum und Maximum
<code>diff(range())</code>	Spannweite
<code>cov()</code>	Stichproben-Kovarianz
<code>cor()</code>	Korrelation (Spearman, Bravais–Pearson)
<code>density()</code>	Kerndichteschätzer
<code>ecdf()</code>	Empirische Verteilungsfunktion

Nützliche Funktionen – Übersicht

Funktion	Beschreibung
<code>cumsum()</code>	Kumulierte Summe
<code>cumprod()</code>	Kumuliertes Produkt
<code>gamma()</code>	Gamma-Funktion: $\Gamma(n) = (n-1)!$
<code>beta()</code>	Betafunktion: $B(a, b) = \frac{\Gamma(a)\Gamma(b)}{\Gamma(a+b)}$
<code>sort()</code>	Sortieren eines Vektors
<code>rank()</code>	Bestimmung der Ränge
<code>factorial()</code>	Fakultät
<code>choose()</code>	Binomialkoeffizient $\binom{n}{k}$
<code>unique()</code>	Entfernung von Duplikaten
<code>duplicated()</code>	Indizierung von Duplikaten

Nützliche Funktionen – Diskretisieren

Diskretisierung einer (quasi-)stetigen Variable:

```
> x <- c(1.3, 1.5, 2.5, 3.8, 4.1, 5.9, 7.1, 8.4, 9.0)
> xdiscrete <- cut(x, breaks = c(-Inf, 2, 5, 8, Inf))
> is.factor(xdiscrete)
[1] TRUE
> xdiscrete
[1] (-Inf,2] (-Inf,2] (2,5]      (2,5]      (2,5]      (5,8]      (5,8]
[8] (8, Inf] (8, Inf]
Levels: (-Inf,2] (2,5] (5,8] (8, Inf]
> table(xdiscrete)
xdiscrete
(-Inf,2]      (2,5]      (5,8] (8, Inf]
         2         3         2         2
```

(Pseudo-) Zufallszahlen

- Statistik verwendet Zufallszahlen
- Generierbar nur durch deterministischen Algorithmus (Generator)
⇒ Pseudo-Zufallszahlen
- Problem: Vergleichbarkeit von zufälligen Ergebnissen
- Lösung: Zufallszahlengenerator initialisieren mit `set.seed()`

```
> set.seed(12345)
> rnorm(n = 2)
[1] 0.5855288 0.7094660
> rnorm(n = 2)
[1] -0.1093033 -0.4534972
> set.seed(12345)
> rnorm(n = 2)
[1] 0.5855288 0.7094660
```

Verteilungen und Zufallszahlen

Funktionen zur Berechnung von Dichten, Verteilungsfunktionen, Quantilen und Erzeugung von (Pseudo-) Zufallszahlen bekannter Verteilungen

Funktionsnamen-Schema

Anfangsbuchstabe	Art der Funktion
d	Dichte (density)
p	Verteilungsfunktion (probability)
q	Quantilsfunktion (quantiles)
r	Zufallszahl (random number)

Übersicht Modellverteilungen

Funktionsende	Verteilungsname
<code>beta</code>	Beta-Verteilung
<code>binom</code>	Binomial-Verteilung
<code>cauchy</code>	Cauchy-Verteilung
<code>exp</code>	Exponential-Verteilung
<code>gamma</code>	Gamma-Verteilung
<code>geom</code>	Geometrische-Verteilung
<code>hyper</code>	Hypergeometrische-Verteilung
<code>lnorm</code>	Log-Normal-Verteilung
<code>norm</code>	Normal-Verteilung
<code>pois</code>	Poisson-Verteilung
<code>unif</code>	Gleich-/ Rechtecks-Verteilung
<code>mvnorm</code>	Multivariate Normal-Verteilung (package <code>mvtnorm</code>)

Übersicht Prüfverteilungen

Funktionsende	Verteilungsname
<code>chisq</code>	χ^2 -Verteilung
<code>f</code>	F -Verteilung
<code>signrank</code>	Verteilung der Wilcoxon Vorzeichen-Rangsummen (1 Stichprobe)
<code>t</code>	t -Verteilung
<code>wilcox</code>	Verteilung der Wilcoxon Rangsummen (2 Stichproben)

Funktionsnamen-Schema – Beispiel Std.-NV

- Dichte der $N(0, 1)$ –Verteilung an der Stelle $x = 0$:
(theoretisch: $1/\sqrt{2\pi}$)

```
> dnorm(x = 0)
```

```
[1] 0.3989423
```

```
> 1 / sqrt(2 * pi)
```

```
[1] 0.3989423
```

- Verteilungsfunktion der $N(0, 1)$ –Verteilung an der Stelle q :
 $\Phi(q) = P(X \leq q)$

```
> pnorm(q = 0)
```

```
[1] 0.5
```

```
> pnorm(q = 1.96)
```

```
[1] 0.9750021
```

Funktionsnamen-Schema – Beispiel Std.-NV

- p -Quantil der $N(0, 1)$ -Verteilung z_p :

```
> qnorm(p = 0.95)
[1] 1.644854
```

- Stichprobe vom Umfang $n = 5$ aus $N(0, 1)$ -Verteilung

```
> X <- rnorm(n = 5)
> X
[1] -0.1093033 -0.4534972  0.6058875 -1.8179560  0.6300986
```

Die `sample()` Funktion

Ziehen einer Stichprobe

- mit festem Umfang (Argument `size`)
- aus endlich diskreten Mengen (Argument `x`)
- mit Zurücklegen (Argument `replace = TRUE`)
- oder ohne Zurücklegen (Argument `replace = FALSE`)
- und optional mit bestimmten Wahrscheinlichkeiten (Argument `prob`).

Argument `replace` ist auf `FALSE` voreingestellt.

Die `sample()` Funktion – Beispiele

- Ziehen mit Zurücklegen aus einer Gleichverteilung über $\{1, 2, 3, 4, 5\}$:

```
> sample(x = c(1, 2, 3, 4, 5), size = 10, replace = TRUE)
[1] 2 3 2 3 1 5 3 2 5 4
```

- Ziehen mit Zurücklegen aus einer vorgegebenen Verteilung (`prob` gesetzt):

```
> zmv <- sample(x = c(1, 2, 3, 4, 5), size = 1000, replace = TRUE,
+               prob = c(0.1, 0.1, 0.4, 0.3, 0.1))
> table(zmv)

zmv
 1   2   3   4   5
98 112 383 306 101
```

Aufgaben

1. Erzeugen Sie Stichproben aus verschiedenen Verteilungen (Poisson, Binomial, χ^2 , Exponential) mit verschiedenen Parametern und den Stichprobenumfängen $n = 20$, $n = 50$, $n = 100$ und $n = 1000$.
2. Berechnen Sie jeweils das arithmetische Mittel und vergleichen Sie es mit dem dazugehörigen theoretischen Erwartungswert.
3. Permutieren Sie den Vektor der Großbuchstaben **LETTERS**.