LMU LUDWIG-MAXIMILIANS-UNIVERSITÄT MÜNCHEN

Statistische Software (R)

Paul Fink, M.Sc.

Institut für Statistik
Ludwig-Maximilians-Universität München

Pseudo Zufallszahlen, Dichten, Verteilungsfunktionen, etc.

Übersicht Statistik-Funktionen

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

Nützliche Funktionen

- Sortieren eines Vektors:

```
> x <- c(1, 3, 2, 5)
> sort(x)
[1] 1 2 3 5
> sort(x, decreasing = TRUE)
[1] 5 3 2 1
> sort(c("Morgen", "Mittag", "Nachmittag", "Abend", "Nacht"))
[1] "Abend"      "Mittag"     "Morgen"     "Nachmittag" "Nacht"
```

- Bestimmung der Ränge:

```
> x <- c(1, 3, 2, 5, 2)
> rank(x)
[1] 1.0 4.0 2.5 5.0 2.5
```


Nützliche Funktionen

- Indizierung mehrfach vorkommender Werte in einem Vektor:

```
> x <- c(1, 3, 2, 5, 2)
> duplicated(x)
[1] FALSE FALSE FALSE FALSE  TRUE
```

- Entfernung von Duplikaten (z.B. Bestimmung aller vorkommenden Merkmalsausprägungen in einer Stichprobe):

```
> x <- c(1, 3, 2, 5, 2)
> unique(x)
[1] 1 3 2 5
```

Nützliche Funktionen

- 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
```


Nützliche Funktionen

- Kumulierte Summe und Produkt:

```
> x <- c(1, 3, 2, 5)
> cumsum(x)          # 1, 1+3, 1+3+2, 1+3+2+5
[1]  1  4  6 11
> cumprod(x)         # 1, 1*3, 1*3*2, 1*3*2*5
[1]  1  3  6 30
```

- Fakultät:

```
> factorial(5)
[1] 120
```

- Binomialkoeffizient $\binom{n}{k}$:

```
> choose(4, 2)
[1] 6
```


Nützliche Funktionen

- Gammafunktion:

  Für natürliche Zahlen n gilt: $\Gamma(n) = (n-1)!$

```
> c(gamma(5), factorial(4))
[1] 24 24
> c(gamma(0.5), sqrt(pi))
[1] 1.772454 1.772454
```

- Betafunktion:

  $B(a,b) = \frac{\Gamma(a)\Gamma(b)}{\Gamma(a+b)}$

```
> c(beta(5, 3), gamma(5) * gamma(3) / gamma(5 + 3))
[1] 0.00952381 0.00952381
```


Verteilungen und Zufallszahlen

Funktionen zur Berechnung von Dichten, Verteilungsfunktionen, theoretischen Quantilen und Erzeugung von (Pseudo–) Zufallszahlen

Funktionsnamen-Schema

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

Funktionsnamen-Schema – Beispiel Std.-NV

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

```
> c(dnorm(x = 0), 1 / sqrt(2 * pi))
[1] 0.3989423 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))
[1]  0.08490976 -0.09564817  0.09662517 -1.14158372 -1.66845847
```


Übersicht Modellverteilungen

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

Übersicht Prüfverteilungen

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

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 5 2 5 4 2 5 5 4
```

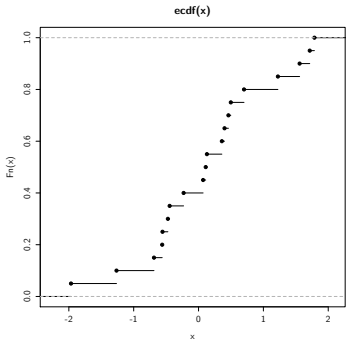
- Ziehen mit Zurücklegen aus einer vorgegebenen Verteilung (`prob` gesetzt):

```
> zmzv <- 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(zmzv)
zmzv
  1   2   3   4   5
 91  92 388 332  97
```


Empirische Verteilungsfunktion

Beispiel Standardnormalverteilung:

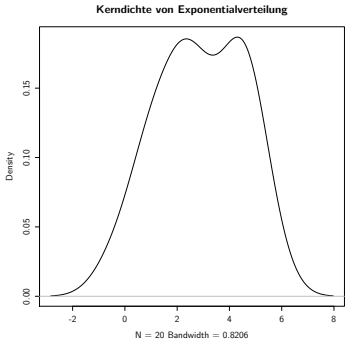
```
> set.seed(123)
> x <- rnorm(n = 20)
> plot(ecdf(x))
```



Kenrdichteschätzung

Beispiel Normalverteilung mit $\mu = 3$ und $\sigma^2 = 4$:

```
> kernds <- density(rnorm(n = 20, mean = 3, sd = 2))
> plot(kernds, main = "Kerndichte von Exponentialverteilung")
```



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 $n = 1000$. Visualisieren Sie die die standardisierten Summen mittels Kerndichteschätzung.
2. Zeigen Sie, dass das Vorgehen wie in 1. für die Cauchy–Verteilung nicht klappt.