

Statistische Funktionen: Verteilungen

- In R stehen die gängigsten Verteilungsfunktionen jeweils in 4 Ausprägungen zur Verfügung:

- **Dichte:** **d**verteilung
- **Verteilung:** **p**verteilung
- **Quantil:** **q**verteilung
- **Stichprobe:** **r**verteilung

- Beispiele:

```
> dbinom(0:2, 2, 0.5)
[1] 0.25 0.50 0.25

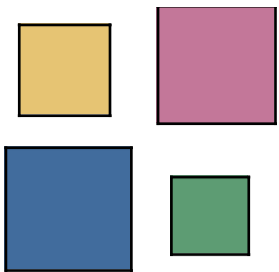
> pbinom(0:2, 2, 0.5)
[1] 0.25 0.75 1.00

> rbinom(10, 2, 0.5)
[1] 2 1 1 2 0 2 2 1 1 2

> dnorm(0) * sqrt(2 * pi)
[1] 1
```

- Verteilungen:

- | | | | |
|-------------------|------------------|-----------------|-----------------|
| – beta | Beta | – binom | Binomial |
| – cauchy | Cauchy | – exp | Exponential |
| – gamma | Gamma | – geom | Geometrisch |
| – hyper | Hypergeometrisch | – lnorm | Log-Normal |
| – norm | Normal | – pois | Poisson |
| – unif | Gleich | – mvnorm | Multivariate NV |
| <hr/> | | | |
| – chisq | chi-quadrat | – f | F |
| – t | t | – wilcox | Wilcoxon (2-SP) |
| – signrank | Wilcoxon (1 SP) | | |



Statistische Funktionen: Sampling

- Mittels der Funktion `sample()` kann man zufällig, gleichverteilte Stichproben aus einem Vektor ziehen
 - Dabei kann mit oder ohne Zurücklegen gezogen werden, Parameter `replace = [T|F]`
 - Der Parameter `size` bestimmt, wie viele Elemente gezogen werden sollen
 - Ein optionaler Vektor `prob` kann eine von der Gleichverteilung abweichende Verteilung angeben

- Beispiele:

```
> sample(x=1:3, 10)
```

```
Error in sample(length(x), size, replace, prob) :  
cannot take a sample larger than the population  
when 'replace = FALSE'
```

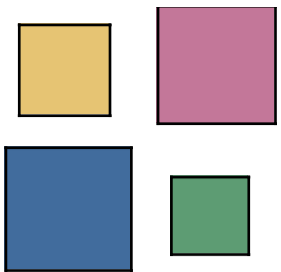
```
> sample(x=1:3, 10, rep=T)
```

```
[1] 1 1 3 1 1 3 3 2 3 3
```

```
> sample(0:2, 10, prob=c(0.25, 0.50, 0.25), rep=T)
```

```
[1] 1 0 0 0 1 1 0 1 2 1
```

- `set.seed(n)` setzt einen eindeutigen Fixpunkt für den Zufallszahlengenerator.



Statistische Funktionen: Sortieren

- Die Rang Statistik (order statistics) $x_{(1)}, \dots, x_{(n)}$ spielt in der Statistik eine zentrale Rolle. Viele elementare Statistiken wie Median oder Quantile, aber auch komplexere Testverfahren, beruhen auf der sortierten Stichprobe $x_{(1)}, \dots, x_{(n)}$.

```
> data(Animals)
```

```
> Animals[1:5,]
```

	body	brain
Mountain beaver	1.35	8.1
Cow	465.00	423.0
Grey wolf	36.33	119.5
Goat	27.66	115.0
Guinea pig	1.04	5.5

```
> sort(Animals$body)
```

```
[1] 0.023 0.120 0.122 0.280 1.04 1.35 2.50
[8] 3.300 6.800 10.000 27.660 35.00 36.33 52.16
[15] 55.500 62.000 100.000 187.100 192.00 207.00 465.00
[22] 521.000 529.000 2547.000 6654.000 9400.00 11700.00 87000.0
```

```
> order(Animals$body)
```

```
[1] 20 19 27 25 5 1 21 11 17 10 4 18 3 24 22 14 23 8 28
[20] 13 2 9 12 7 15 16 6 26
```

```
> rank(Animals$body)
```

```
[1] 6 21 13 11 5 27 24 18 22 10 8 23 20 16 25 26 9 12 2
[20] 1 7 15 17 14 4 28 3 19
```

```
> Animals[order(Animals$body)[1:5],]
```

	body	brain
Mouse	0.023	0.4
Golden hamster	0.120	1.0
Mole	0.122	3.0
Rat	0.280	1.9
Guinea pig	1.040	5.5