

```
library(pander)
library(magrittr)
library(tidyverse)
```

```
## -- Attaching packages ----- ti
## v ggplot2 3.2.1      v purrr  0.3.3
## v tibble  2.1.3      v dplyr  0.8.4
## v tidyr   1.0.2      v stringr 1.4.0
## v readr   1.3.1      v forcats 0.4.0

## -- Conflicts ----- tidyvers
## x tidyr::extract() masks magrittr::extract()
## x dplyr::filter() masks stats::filter()
## x dplyr::lag() masks stats::lag()
## x purrr::set_names() masks magrittr::set_names()
```

Eine häufige Hürde ist die Übertragung von Messdaten in eine Form, welche ohne viel Umstrukturierung in der R-Umgebung genutzt werden kann. Das Verständnis für sinnvolle Strukturen ist jedoch auch von der Anwendung abhängig, welche sonst zur Auswertung verwendet wird, in Excel sind Datenstrukturen häufig verschachtelt aber für das Rechnen in den Excel-Sheets durchaus sinnvoll strukturiert. Sobald jedoch ein solcher Datensatz in der R-Umgebung weiter bearbeitet wird, kommt es zwangsläufig zu Fehlinterpretationen. Bei der Aufnahme von Messdaten ist die Empfehlung alles so *basic* wie möglich zu halten. Ein Beispiel dafür ist die Messung von Absorptionen mit vier technischen Replikaten je Messung. Wenn diese mehrfach wiederholt wird wobei eine Änderung am System vorgenommen wird, so ist zu beachten, dass alle Mehrfachbestimmungen so gruppiert werden, dass sie eindeutig einer Messung zugeordnet werden können:

Messung 1	Messung 2	Messung n
Wert 1	Wert 1	Wert 1
Wert 2	Wert 2	Wert 2
Wert 3	Wert 3	Wert 3
Wert 4	Wert 4	Wert 4

Wenn diese Technischen Replikate nun in R importiert werden, bleibt die Struktur erhalten. Im hier dargestellten Fall handelt es sich um den Optimalfall:

- jede Variable verfügt über eine Spalte
- jede Beobachtung wird in einer Zeile einer Variable zugeordnet.

Das oben dargestellte Beispiel kann in einem tibble leicht dargestellt werden:

```
Messdaten <- tibble(
  "Messung 1" = c("Wert 1", "Wert 2", "Wert 3", "Wert 4"),
  "Messung 2" = c("Wert 1", "Wert 2", "Wert 3", "Wert 4"),
  "Messung n" = c("Wert 1", "Wert 2", "Wert 3", "Wert 4")
)
```

Nun kann es aber gut sein, dass zusätzliche Informationen mit einfließen, eventuell die Temperatur, die Viskosität oder eine anderer Parameter. Auf einem Excel-Spread-Sheet würde dafür meist eine neue Spalte oder Zeile dafür einführen. In der R-Umgebung entspricht das einer Liste, denn eine Liste kann ja verschiedene Informationen enthalten. Eine solche Liste könnten so aussehen:

```
Datensatz <- list(
  temp. <- c("Temperatur 1", "Temperatur 2", "Temperatur 3", "Temperatur 4"),
  Messdaten,
  "Viskosität" <- c("Viskosität 1", "Viskosität 2", "Viskosität 3", "Viskosität 4")
)
pander(Datensatz)
```

- Temperatur 1, Temperatur 2, Temperatur 3 and Temperatur 4

Messung 1	Messung 2	Messung n
Wert 1	Wert 1	Wert 1
Wert 2	Wert 2	Wert 2
Wert 3	Wert 3	Wert 3
Wert 4	Wert 4	Wert 4

-
- Viskosität 1, Viskosität 2, Viskosität 3 and Viskosität 4

Als Beispiel ein realer Datensatz:

```
library(readxl)
Messwerte <- list(
  "Diese Liste enthält die Messwerte der ursprünglichen Tabelle, nur sauberer für R",
  Progesteron.Standard <- read_excel("Progesteron.xlsx", sheet = 2),
  Progesteron.Proben.1 <- read_excel("Progesteron.xlsx", sheet = 3),
  Progesteron.Proben.2 <- read_excel("Progesteron.xlsx", sheet = 4)
)
pander(Messwerte)
```

- Diese Liste enthält die Messwerte der ursprünglichen Tabelle, nur sauberer für R

Table 3: Table continues below

Konzentration	0	0.1	0.2	0.5	1	2	5	10
n1	3.121	3.194	2.588	2.644	1.671	1.156	0.672	0.629
n2	2.886	3.071	2.699	2.508	1.75	1.294	0.619	0.517
n3	3.082	2.794	2.296	3.163	1.942	1.002	0.653	0.527

	20	50
	0.257	0.195
	0.208	0.222
	0.31	0.231

-

Testperson	n1	n2
1	2.552	2.623
2	2.086	1.911
3	2.619	2.378

Testperson	n1	n2
4	1.765	2.103
5	1.134	1.236
17	0.71	1.012
7	1.748	1.593
8	2.002	1.663
9	1.4	1.344
10	2.084	2.171
11	1.697	1.906
12	1.552	1.834
13	1.326	1.792
14	1.553	1.586
15	2.66	2.13
16	1.221	2.463

•

verd	20	100
n1	0.399	0.755
n2	0.41	1.699
n3	0.404	0.71
n4	0.377	0.796

•

In diesem Datensatz wurden die Werte in einer, für Excel sinnvoll wirkenden Art zusammengestellt, wobei uns das bei der Weiterverarbeitung in der R-Umgebung eher hindern wird. Es empfiehlt sich, den Datensatz innerhalb von Excel nun soweit zu ändern, dass er leichter in eine Liste gewandelt werden kann, also nach der bereits beschriebenen Struktur abzubilden.