# Einführung in R und RStudio

## [Tag 1 & 2]

Miguel Alvarez

18. Oktober 2022

# Der Kurs

- **Termin 1 & 2**
  - Grundlagen
  - Datentypen
- **Termin 3 & 4**
  - Objekten
  - Lesen und Schreiben
- **Termin 5 & 6**
  - Statistiken
  - Grafiken
- **Termin 7 & 8**
  - Automatisieren
  - Erstellen von Dokumenten
  - Abschluss

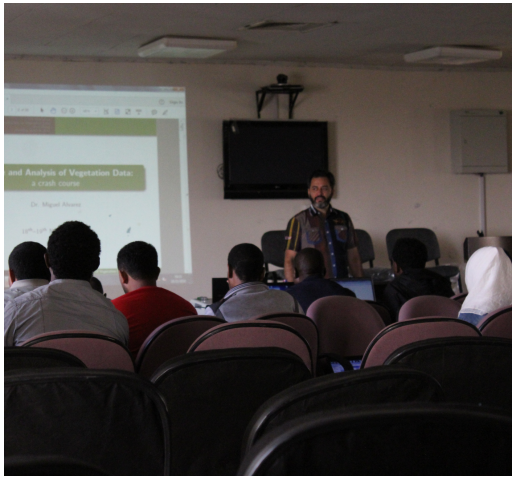
# Der Kurs

18:00 – 20:30

Pause 19:15 – 19:30

- Methoden
  - Vortrag
  - Life-Codierung
  - Übungen

https://kamapu.github.io/RKurs-VHS-2022/



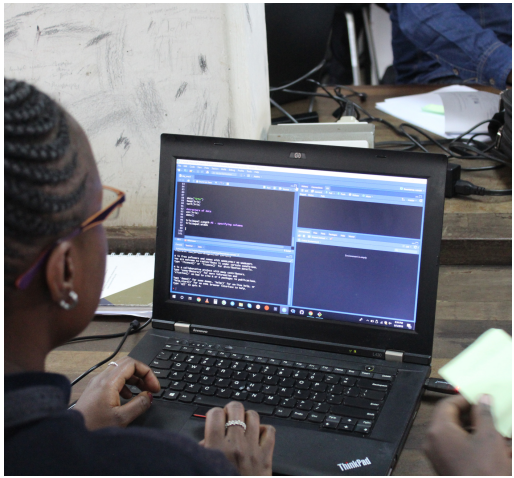
# Der Kurs

## Miguel Alvarez

- Vegetationsökologist
- Begeisterter R-Nutzer
- R-Programmierer
  - Umwelt und Biodiversität
  - Geographische Informationssystheme
  - Datenbanken
  - Reproduzierbarkeit

## Die Teilnehmer?

- Warum R?
- Erwartung von dem Kurs



## Geschichte

- S (1975)
  - S-PLUS
  - TIBCO Spotfire S+
- R (1992)

**R** ist die kostenlose Alternative zu **S**

**R**oss Ihaka
**R**obert Gentleman

## Was ist R?

- Programmiersprache
  - Open Source (offene Quelle)
  - Freeware (kostenlos)
- Statistische Umgebung
  - Interface
  - Terminal
  - *Editoren*

https://www.r-project.org/

## Was ist R?

- Schwerpunkt
  - Mathematik
  - Grafiken
- Kommandozeilen
- REPL (Read-Eval-Print-Schleife)

```
(5 + 100) * 25
```

```
## [1] 2625
```

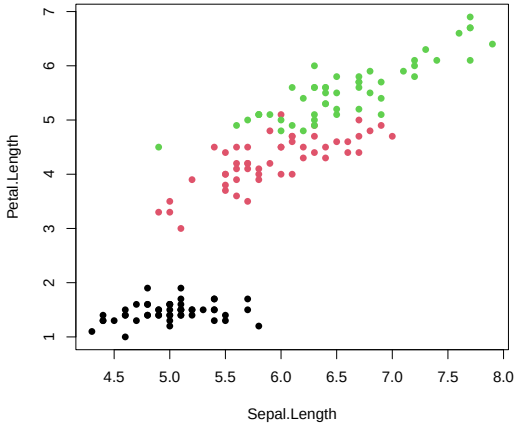
```
50/10
```

```
## [1] 5
```

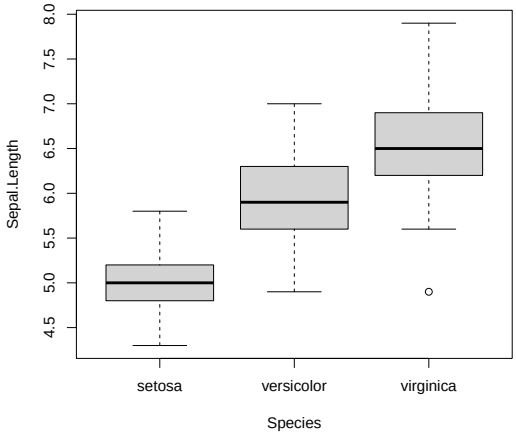
```
60 > 15
```

```
## [1] TRUE
```

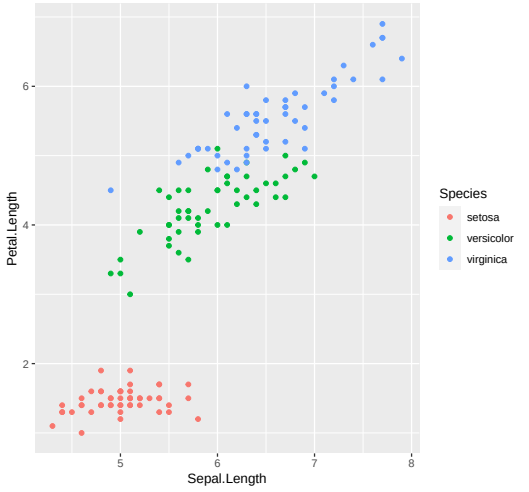
```
plot(iris[ , c("Sepal.Length",
               "Petal.Length")],
     pch = 16, col = iris$Species)
```



```
boxplot(Sepal.Length ~ Species,
    data = iris)
```



```
library(ggplot2)
ggplot(iris,
    aes(x = Sepal.Length,
        y = Petal.Length,
        color = Species)) +
  geom_point()
```



## Warum R?

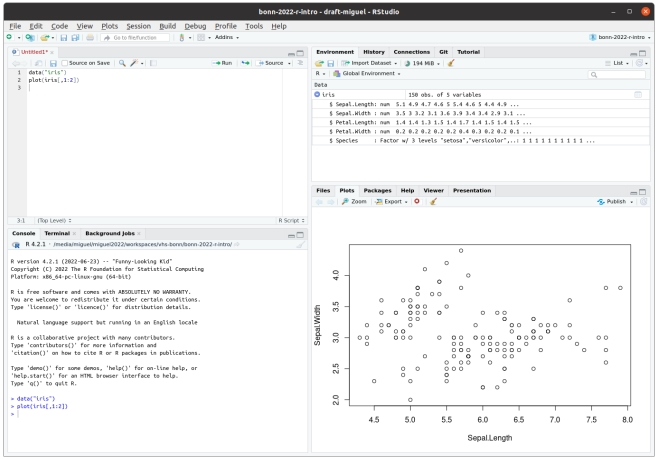
- Kostet nichts
- Steuerung von Analysen
- Skripten
  - Protokolle
  - Reproduzierbarkeit
- Vielseitig

## Jargon

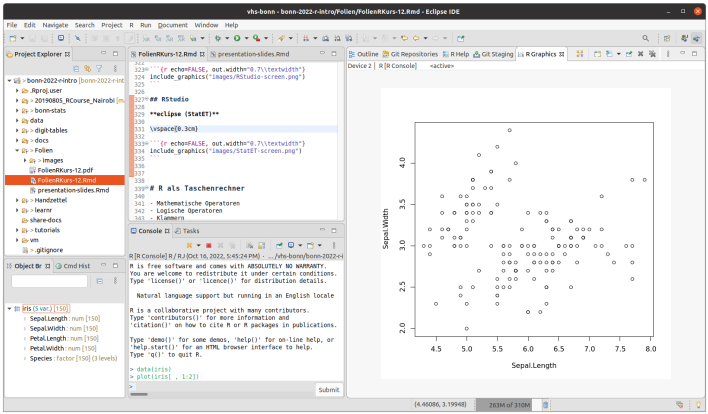
- Konsole
- Sitzung
- Workspace (Environment)
- Arbeitsverzeichnis (working directory)
- Skript
- Paket (Package)

# RStudio

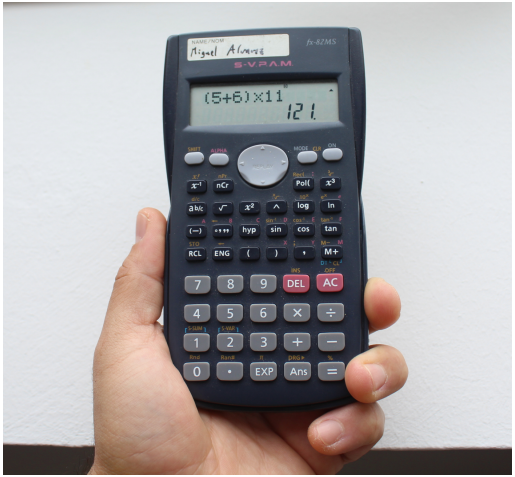
- Text Editor
- IDE (integrated development environment)



## eclipse (StatET)



# R als Taschenrechner




- Mathematische Operatoren
- Klammern

```
5+6*11
```

```
## [1] 71
```

```
(5+6)*11
```

```
## [1] 121
```

```
(5-6)/11
```

```
## [1] -0.09090909
```

# R als Taschenrechner



## Zuweisung

<-

Zuweisung von einem Wert zu einem Objekt.

## Konvention

- Nutze <- anstatt ->
- Reserviere = für Argumente in Funktionen
- Nutze Leezeichen für eine bessere Übersicht in Quellcode
- Nutze Einrückungen für gebrochene Kommandozeilen

# R als Taschenrechner

## Logische Operatoren

```
==
!=
>
>=
<
<=
&
|
%in%
!
any()
all()
```

```
10 > 15
```

```
## [1] FALSE
```

```
10 < 15
```

```
## [1] TRUE
```

# Daten in R

Der Vektor in R ist die grundlegende Datenstruktur

- Länge `length()`
- Typ `class()`
- Evtl. Namen `names()`

```
c(1:10)
```

```
##  [1]  1  2  3  4  5  6  7  8  9 10
```

```
rep(5, times = 10)
```

```
##  [1] 5 5 5 5 5 5 5 5 5 5
```

```
LETTERS[1:5]
```

```
## [1] "A" "B" "C" "D" "E"
```

# Daten in R

## Indexieren

- Eckige Klammern
- Index
  - `integer`
  - `logical` (Bedingung)
  - `character` (Namen)

```
# Mit integer
letters[15]
```

```
##   o
## "o"
```

```
# Mit logischen Werten
letters[!letters %in% c("a", "b", "c")]
```

```
##   d   e   f   g   h   i   j   k   l   m   n
## "d" "e" "f" "g" "h" "i" "j" "k" "l" "m" "n"
##   x   y   z
## "x" "y" "z"
```

```
# Mit Namen
names(letters) <- letters
letters["m"]
```

```
##   m
## "m"
```

## Datentypen

- integer
- numeric
- logical
- factor
- character

## Sonderklassen

- NA
- NaN
- NULL
- Inf
- -Inf
- complex

## Zeit und Datum

# Hilfe!

- Dokumentation
- Forums
- Programfehler melden

Vielen Dank!