

R-Workshop

Nicolas Ruth | HMTM | 26.02.2026

Übungen mit fiktiven Musik & Medien-Studien

DISCLAIMER

Die folgenden Studien sind keine echten Untersuchungen!
Die Themen wurden so gewählt, dass sie zum Themenfeld Musik & Medien passen.
Alle Daten sind Computergeneriert!

A woman with long brown hair and sunglasses is holding a glass of white wine. She is wearing a dark, sequined jacket. In the background, there is a window with a grid pattern.

Werden Musikvideos für Produktplatzierungen genutzt?

- Inhaltsanalyse von 300 Musikvideos
- Variablen:
 - Anwesenheit von Platzierungen
 - Anzahl von Platzierungen
 - Musikgenre

Tanzen Menschen in einer Silent Disco?

- Beobachtung von 300 Personen in einer Silent Disco.
- Variablen
 - Anzahl der tanzenden Personen
 - Hördauer mit Kopfhörern
 - Gründe für die Kopfhörerabnahme

Haben Musiker*innen Lampenfieber vor Interviews?

- Befragung von 100 Musiker*innen
- Variablen:
 - Extraversion (5-Punkte-Likertskala)
 - Erfahrung im Beruf (in Jahren)
 - Selbsteinschätzung Lampenfieber (5-Punkte-Likertskala)

Experiment

Gruppe	Manipulation (UV)	Datenerhebung (AV)
Experimentalgruppe	Freier Eintritt + Spende	Spenderbereitschaft, Interesse
Kontrollgruppe	10 Euro (wird gespendet)	Spenderbereitschaft, Interesse

- Experiment mit 120 Teilnehmenden
- Bedingungen: Freier Eintritt und Spendenaufruf vs. 10 Euro Eintritt, der gespendet wird
- Variablen:
 - Spenderbereitschaft (7-Punkte-Likertskaala)
 - Interesse am Festival (7-Punkte-Likertskaala)

Nützliche Cheatsheets und Zusammenfassungen

Objektbenennungen in R

Snake case (Empfehlung tidyverse)

mein_datensatz

Camel case

meinDatensatz

Pascal case

MeinDatensatz

Dot case

mein.datensatz

Kebab case

Mein-datensatz

Screaming Snake Case

MEIN_DATENSATZ

R Cheatsheet

Zuweisung

`<-` Objekt zuweisen
`=` Zuweisung (in anderen Sprachen üblich)

`> < >= <=` Vergleiche
`&` und
`|` oder
`!` Nicht

Zugriff & Auswahl

`$` Spalte aus Data Frame
`[]` Elemente auswählen
`[,]` Zeilen , Spalten
`[[]]` Ein einzelnes Element

Mathematik

`+ - * / ^`

Klammern

`()` Funktion ausführen
`{ }` Code-Block / Funktion schreiben
`[]` Indexing

Formelschreibweise (Modelle)

`y ~ x` „y erklärt durch x“
`+` weiterer Prädiktor
`*` Interaktion
`:` nur Interaktion

Logische Operatoren / Booleans

`==` gleich
`!=` ungleich

Sonstiges

`#` Kommentar
`,` Argumente trennen
`?` Hilfe zu Funktionen, z. B. `?mean`

RStudio Cheatsheet

Assignment Operator (<-)

Alt + -

Code ausführen

Strg + Enter

Alles ausführen

Strg + Shift + Enter

Letzte Eingaben einsehen

↑

Konsole leeren

Ctrl + L

Neues Script

Ctrl + Shift + N

Pipe (%>%)

Ctrl + Shift + M

tidyverse

Eine Sammlung von R-Paketen mit gemeinsamer „*tidy*“-Philosophie:
klare, lesbare Daten-Workflows für Import, Manipulation, Visualisierung und mehr.

Standardpakete im tidyverse

- **ggplot2** Daten visualisieren (Grafiken nach *Grammar of Graphics*)
- **dplyr** Daten transformieren & wrangeln (filter, select, mutate, etc.)
- **tidyr** Daten *tidy* machen (umformen, wide ↔ long)
- **readr** Daten einlesen (CSV/Tab/Text)
- **purrr** Funktionales Programmieren *map*-Familie
- **tibble** Moderne Daten-Frames (besser lesbar)
- **stringr** Zeichenketten („strings“ oder „char“) bearbeiten
- **forcats** Faktor-/Kategorien-Daten verwalten
- **lubridate** Zeit- und Datums-Handling vereinfachen

Tidyverse-Logik (Pipes)

Grundidee: Daten → Transformation → Ergebnis

Code wird als Abfolge von Schritten gelesen.

Jeder Schritt verändert das Objekt leicht.

Die Pipe

%>% Bedeutung: „Nimm das Ergebnis links und gib es in die Funktion rechts.“

Bsp. ohne Pipe

```
mean(filter(df, gruppe == "A") $wert)
```

Schwer lesbar – von innen nach außen.

Bsp. mit Pipe

```
df %>%
```

```
filter(gruppe == "A") %>%
```

```
summarise(M = mean(wert))
```

Lesbar – von oben nach unten.

Typische Struktur

```
daten %>%
```

```
filter(...) %>%
```

```
select(...) %>%
```

```
group_by(...) %>%
```

```
summarise(...)
```

Warum sinnvoll?

- Klarer Workflow
- Weniger Zwischenspeicher
- Bessere Lesbarkeit
- Gut für reproduzierbare Analysen

Im Kern:

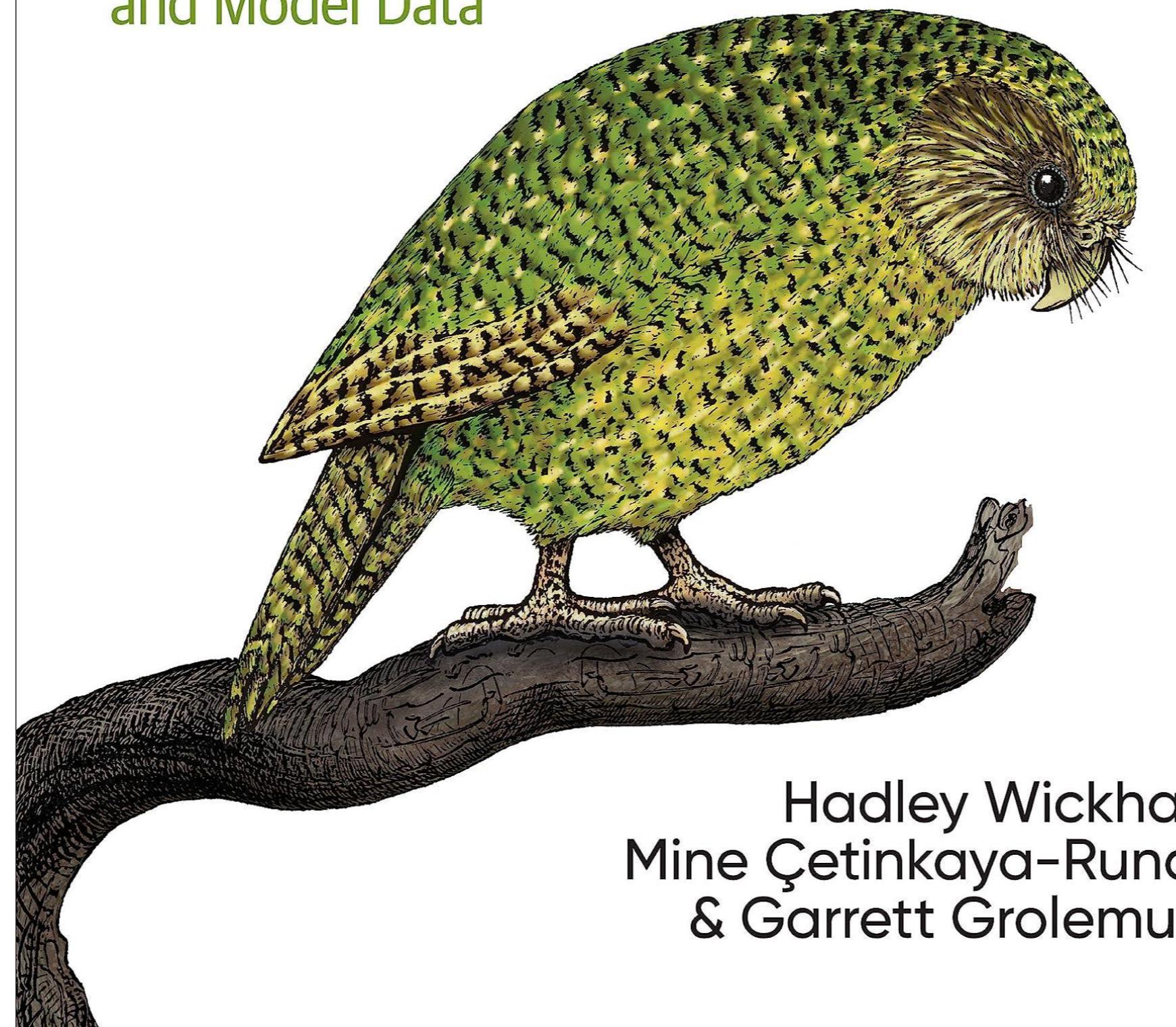
Jede Zeile = ein logischer Schritt in der Datenanalyse.

O'REILLY®

Second
Edition

R for Data Science

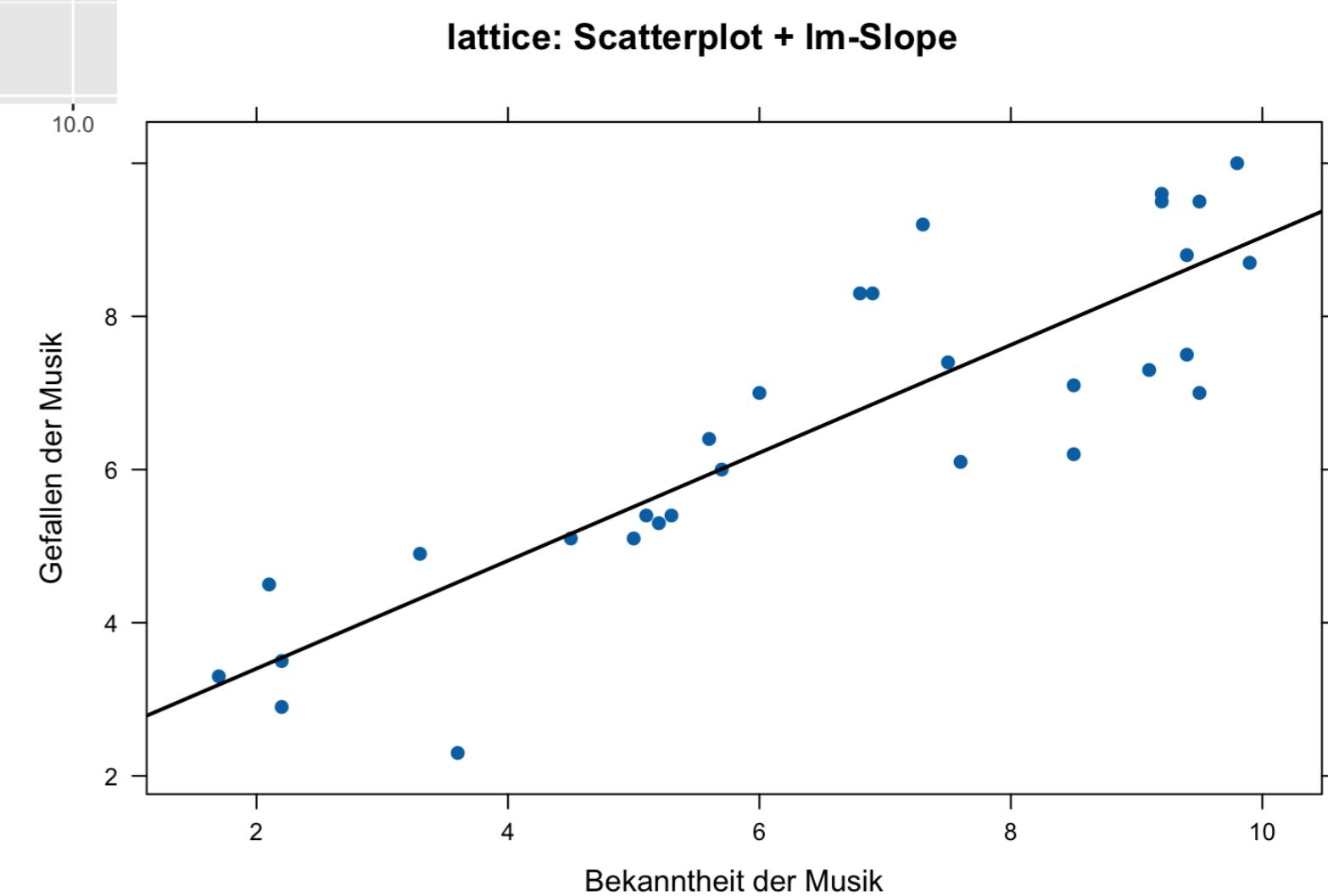
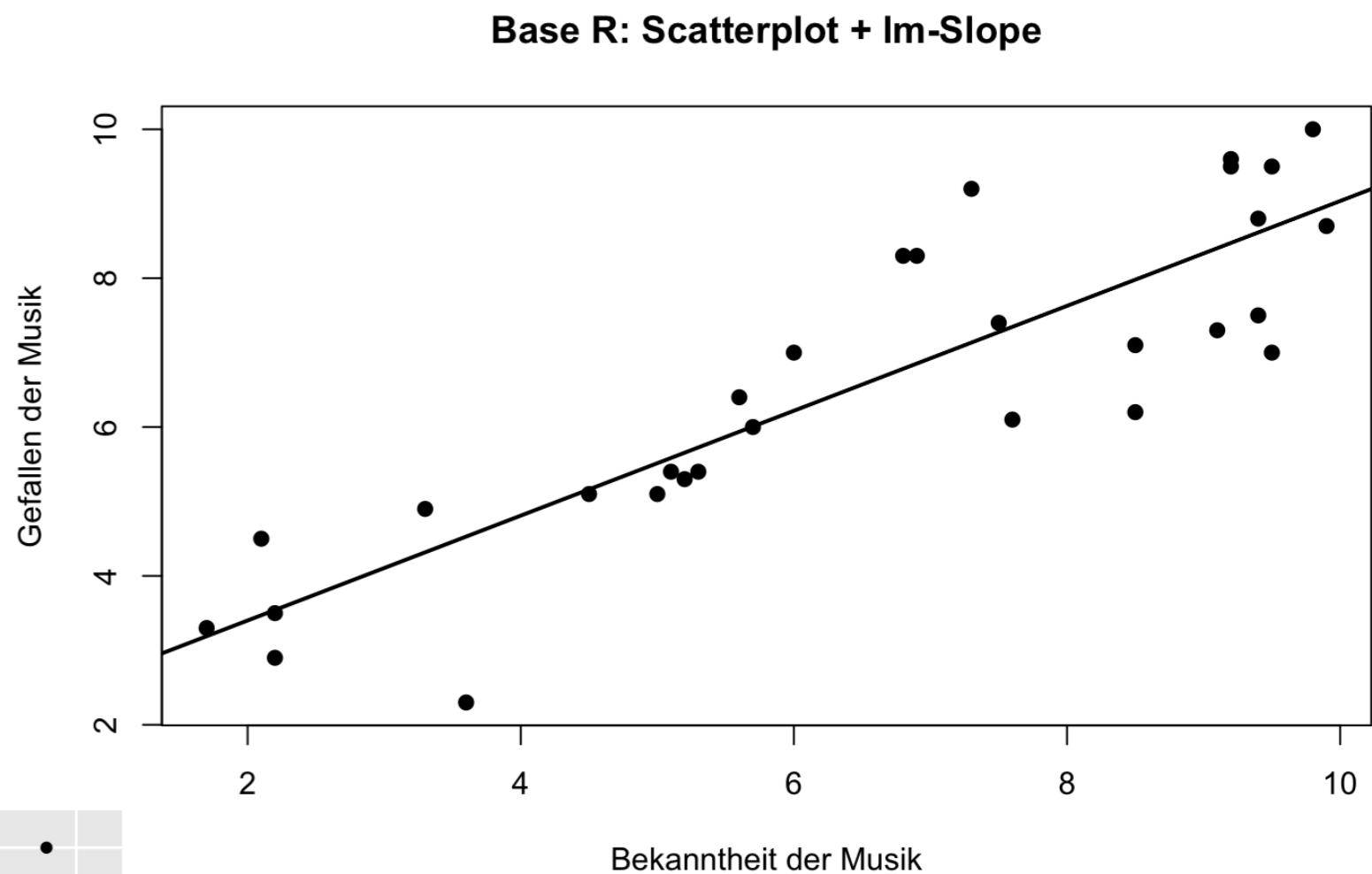
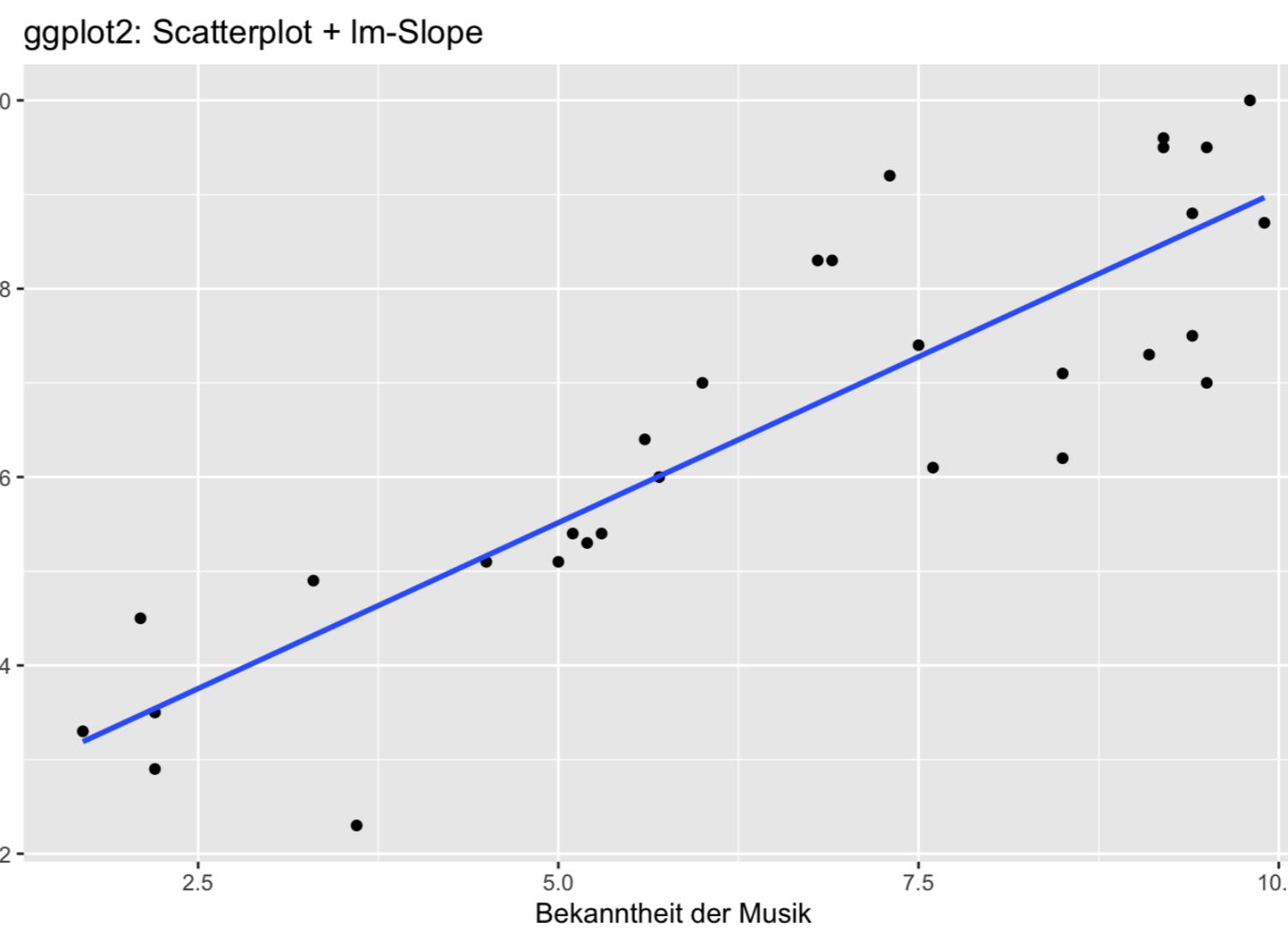
Import, Tidy, Transform, Visualize,
and Model Data



Hadley Wickham,
Mine Çetinkaya-Rundel
& Garrett Grolemund

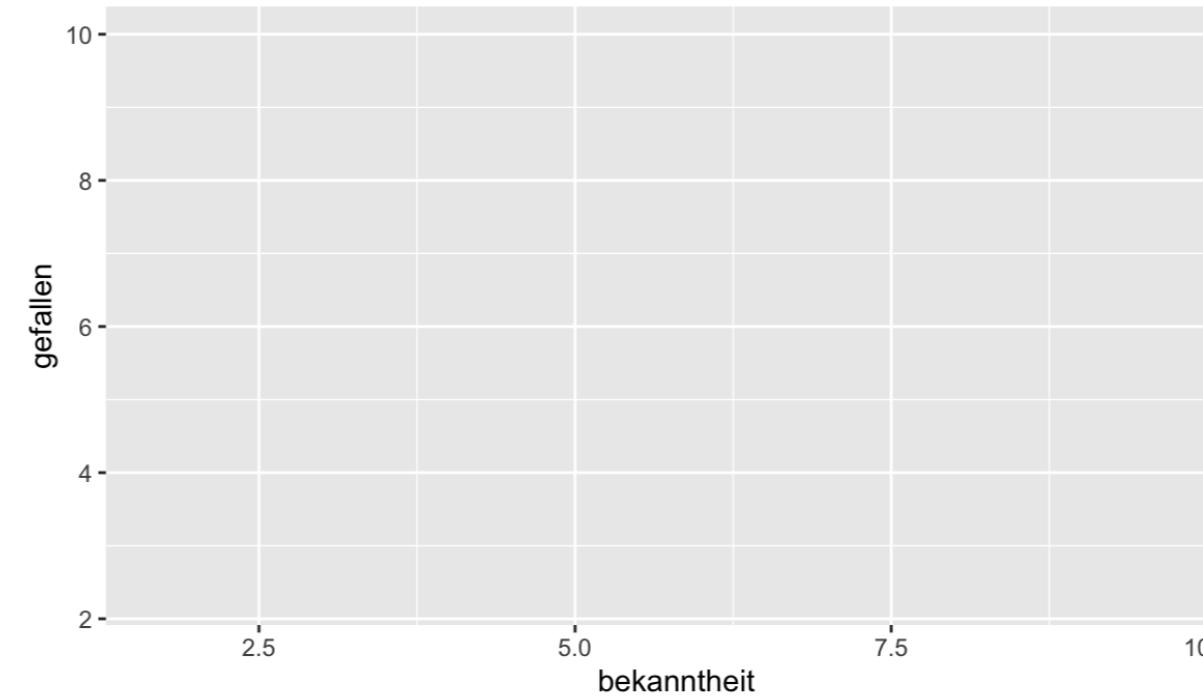
Grafische Analysen

- Base
- GGPlot2
- Lattice

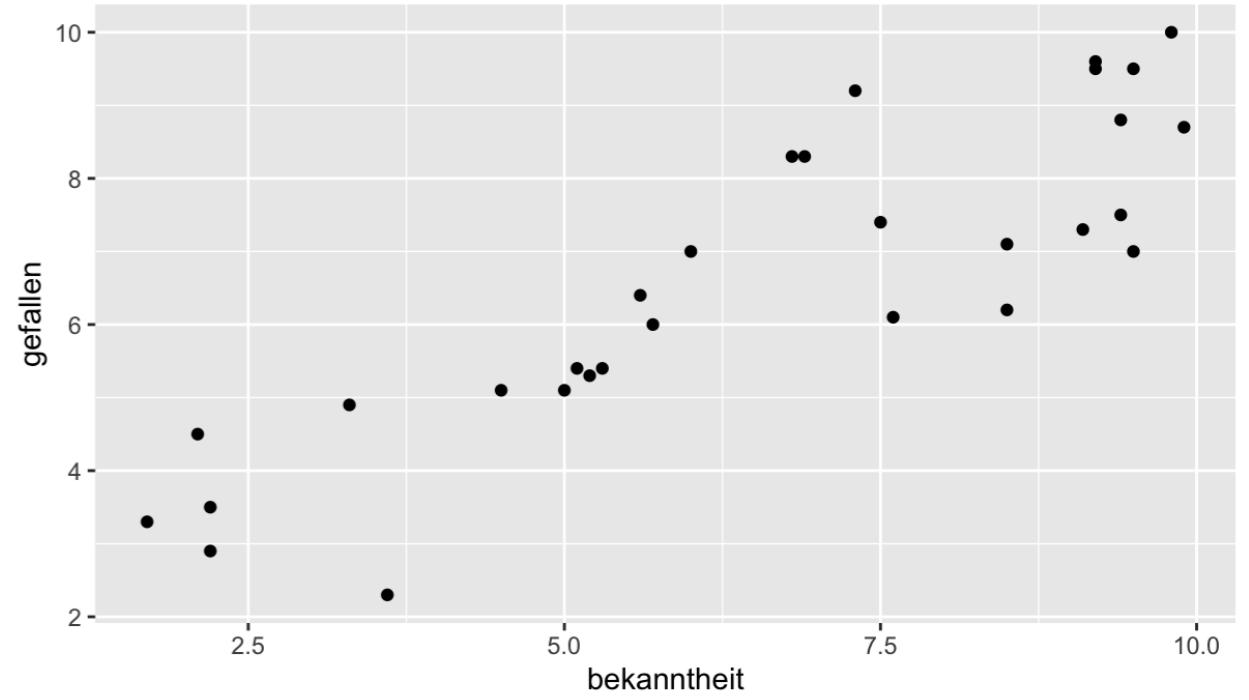


GGPlot-Logik

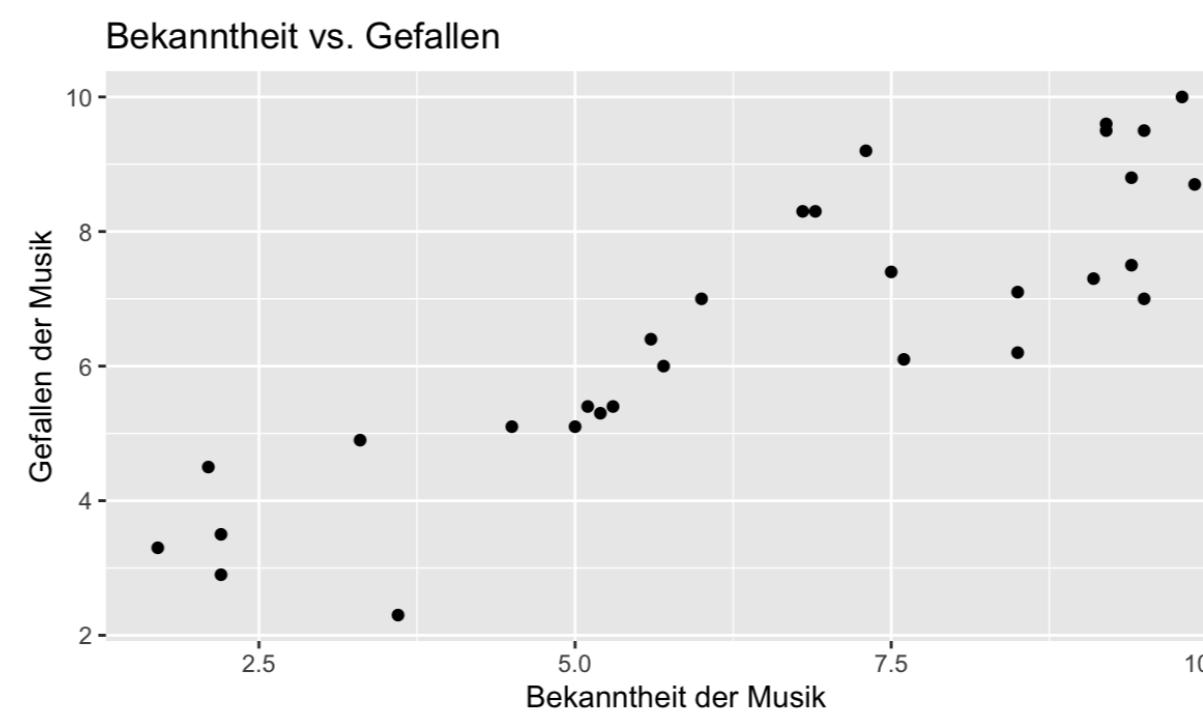
```
korrelation_geschmack_bekanntheit <- ggplot(  
  data = musik,  
  mapping = aes(x = bekanntheit, y = gefallen))  
)
```



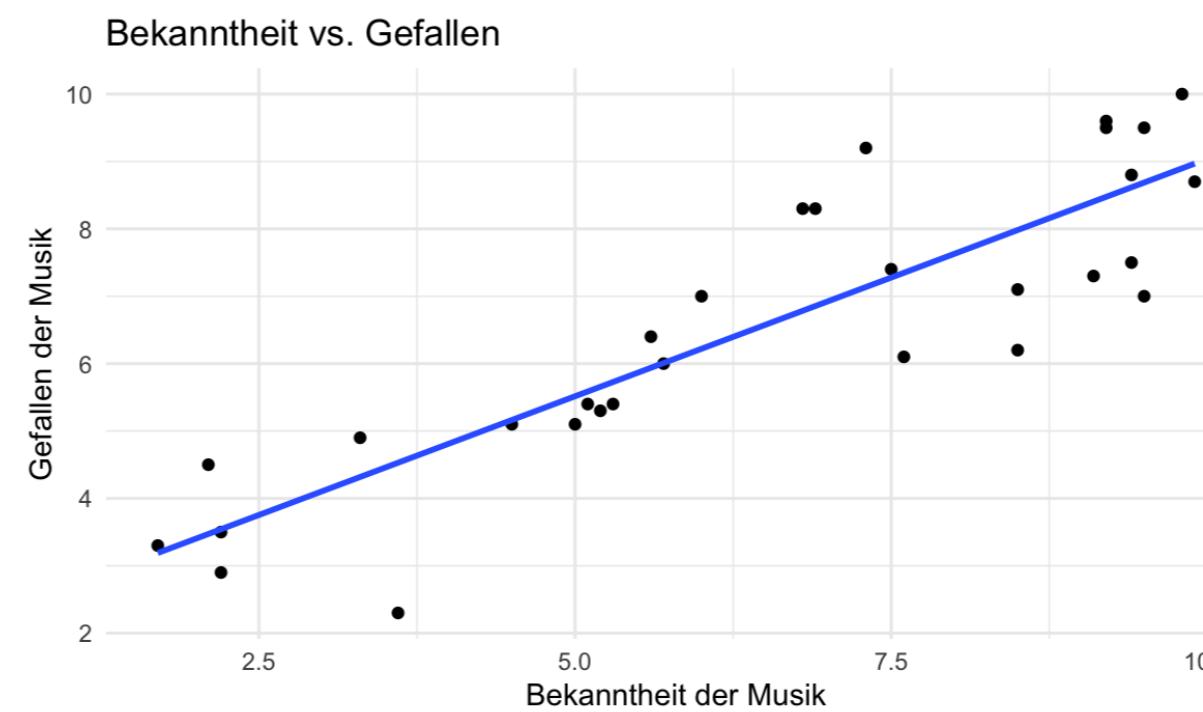
```
korrelation_geschmack_bekanntheit_punkte <-  
  korrelation_geschmack_bekanntheit +  
  geom_point()
```



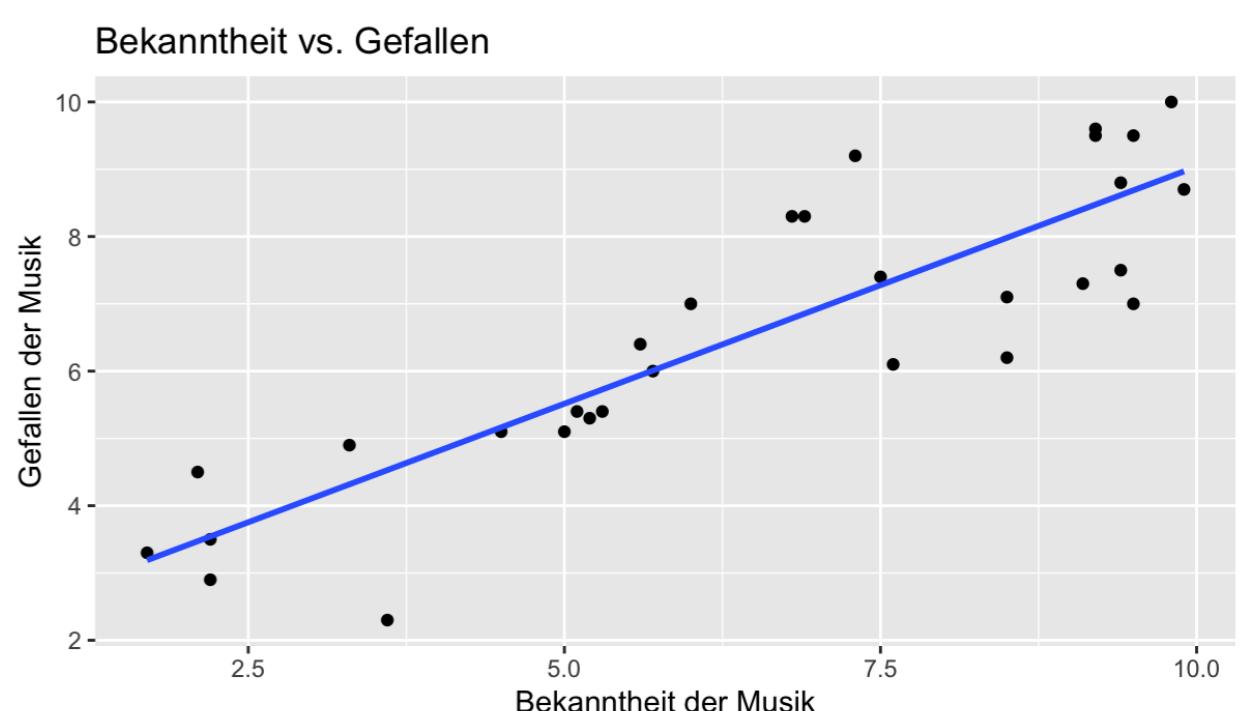
```
korrelation_geschmack_bekanntheit_labels <-  
  korrelation_geschmack_bekanntheit_punkte +  
  labs(  
    title = "Bekanntheit vs. Gefallen",  
    x = "Bekanntheit der Musik",  
    y = "Gefallen der Musik")  
)
```



```
korrelation_geschmack_bekanntheit_final <-  
  korrelation_geschmack_bekanntheit_lm +  
  theme_minimal()
```

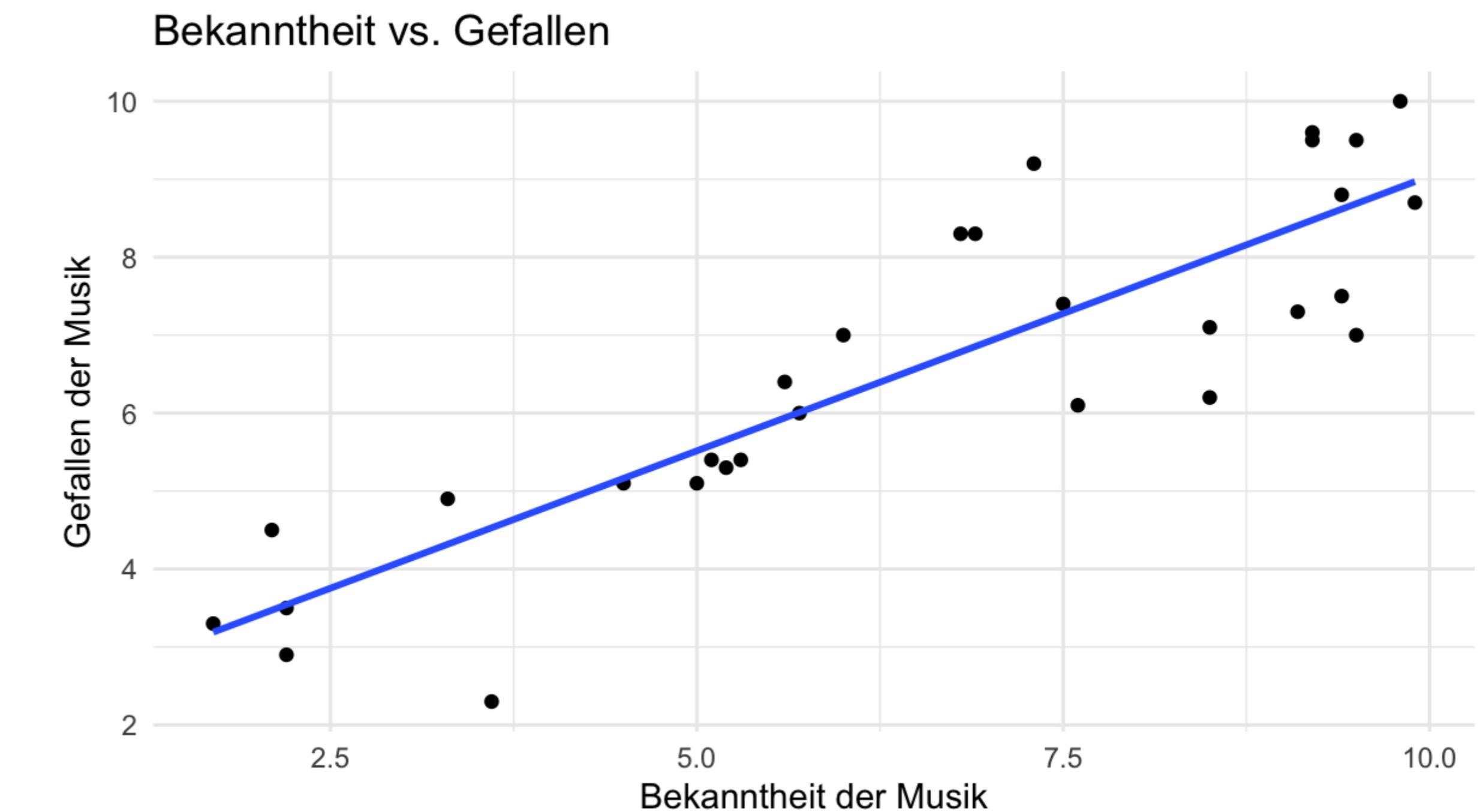


```
korrelation_geschmack_bekanntheit_lm <-  
  korrelation_geschmack_bekanntheit_labels +  
  geom_smooth(method = "lm", se = FALSE)
```



GGPlot-Logik

```
korrelation_geschmack_bekanntheit +  
  geom_point() +  
  geom_smooth(method = "lm", se = FALSE) +  
  labs(title = "Bekanntheit vs. Gefallen",  
       x = "Bekanntheit der Musik",  
       y = "Gefallen der Musik") +  
  theme_minimal()
```



Statistik-Cheatsheet

Deskriptive Statistik

```
mean(x)          # Mittelwert  
median(x)        # Median  
sd(x)           # Standardabweichung  
var(x)           # Varianz  
summary(x)       # 5-Punkte-Summary  
table(x)         # Häufigkeiten  
prop.table(table(x)) # Prozentwerte  
table(x, y)      # Kreuztabelle  
chisq.test(table(x, y))
```

Zusammenhänge

```
cor(x, y)          # Pearson  
cor(x, y, method = "spearman") # Spearman  
cor.test(x, y)      # Signifikanztest  
mod <- lm(y ~ x1 + x2, data = df) # Regression  
summary(mod)
```

Gruppenunterschiede

```
t.test(x, y)          # unabhängige Gruppen  
t.test(x, y, paired = TRUE) # gepaart  
t.test(x ~ gruppe, data = df) # Formel-Schreibweise  
mod <- aov(y ~ gruppe, data = df) # ANOVA  
summary(mod)
```

Packages für weitere Modelle

psych

Explorative Faktorenanalyse
Reliabilität (Cronbach's Alpha)

lavaan

Strukturgleichungsmodelle (SEM)

semPlot

Grafische Darstellung von SEM-Modellen

processR

PROCESS-ähnliche Modelle (Hayes) in R

Lme4 / glm() (Base R)

Lineare & generalisierte Mixed Models
Logistische & Poisson Regression

brms

Bayesianische Regression

For those who care

Swirl()

Karel The Robot

Coursera/Udemy

R for Data Science

Learning statistics using R

www.freecodecamp.org

www.w3schools.com