

Vorbereiten der R-Umgebung

Patrick Zerrer & Katharina Maubach

Table of contents

1	Die R-Studio-Cloud der Universität	1
2	Das Anlegen eines R-Projekts	2
3	Das Working-Dictionary (wd)	4
4	Das Zugreifen auf Dateien und Datensätze	6
5	Die Session	7
6	Hilfe zur Selbsthilfe	7

Bevor wir uns näher mit Datensätzen und deren Analyse beschäftigen, sollten wir uns zunächst einige Grundlagen widmen. Wir beschäftigen uns im Folgenden mit einigen fundamentalen Begriffen, deren Funktionsweisen wir kennen sollten - das erleichtert uns zumindest die Arbeit deutlich und vermeidet wahrscheinlich den ein oder anderen Fehler samt dazugehörigen frustrierten Verfluchen des Kurses und R.

Video

<https://nc.uni-bremen.de/index.php/s/DJGAeBcNing9dQZ/download/%231%20Einf%C3%BChrung%20R.mp4>

Lasst uns Beginnen und Schritt für Schritt die wichtigsten Punkte durchgehen, die ihr zur Vorbereitung eures Analyse-Projekts machen solltet.

1 Die R-Studio-Cloud der Universität

Falls ihr an der Universität Bremen studiert könnt ihr den Service des Zentrum für Multimedia in der Lehre nutzen indem ihr euch [hier](#) mit eurer Uni-Kennung einloggt.



Figure 1: Unser Schreibtisch, Bild generiert von Midjourney

Als nächstes habt ihr die Möglichkeit die für euren Kurs vorgesehene Programmierumgebung auszuwählen. In unserem Fall ist das die “Einführung in quantitative Forschungsdesigns und Datenanalyse”. Mit der Auswahl wird eure eigenen Programmierumgebung eingerichtet, auf dieser könnt ihr die Schaltfläche **RStudio** auswählen.

Mit dem Klick gelangt ihr in R-Studio.

2 Das Anlegen eines R-Projekts

Starten wir zunächst mit dem Anlegen eines sogenannten R-Projekts. RStudio-Projekte ermöglichen es uns alle mit einem Projekt verbundenen Dateien an einem Ort zu speichern. Das umfasst Datensätze, R-Skripte, Ergebnisse, Abbildungen, Berichte usw. Projekte sind bereits in RStudio integriert.

Das gerade aktive RProjekt sehen wir in der rechts oberen Ecke des Nutzer:innen Interfaces von RStudio. Hier können wir durch den Button **New Project** ein neues Projekt anlegen, hierfür folgen wir einfach dem Menüverlauf. Für einen guten, reproduzierbaren Arbeitsablauf sollten alle Analyse Projekte mit der Erstellung eines Projekts beginnen.

In eurem Fall könnt ihr euch direkt alle notwendigen Materialien und Skripte für das erfolgreiche Bearbeiten dieses Kurses in R-Studio laden. Hierfür geht ihr in der rechts oberen Ecke des

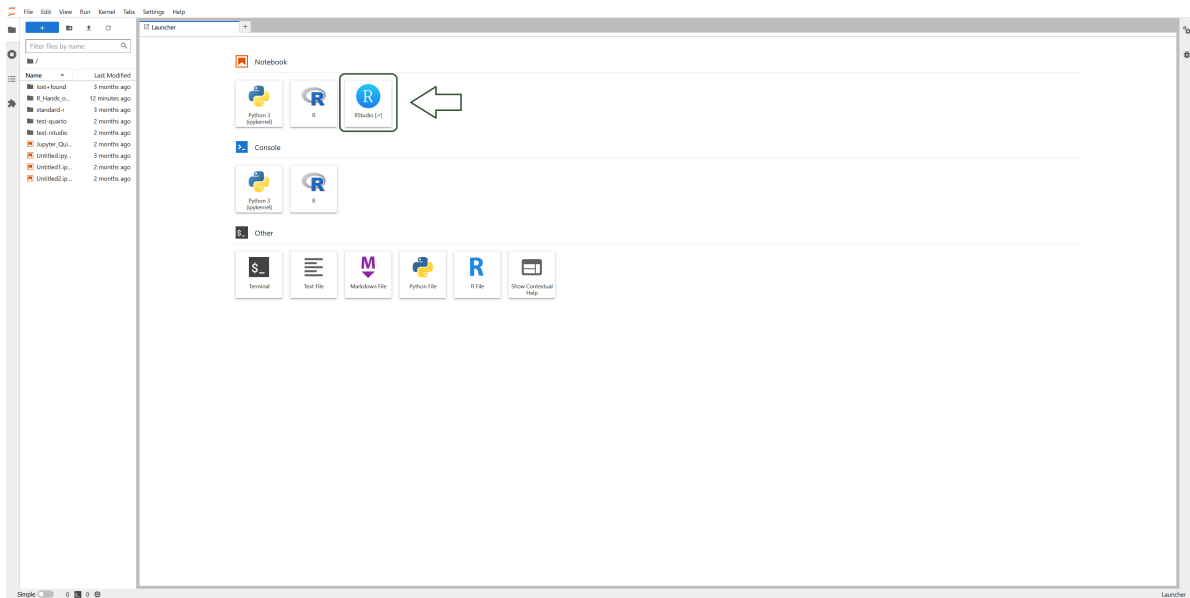


Figure 2: Screenshot RStudio

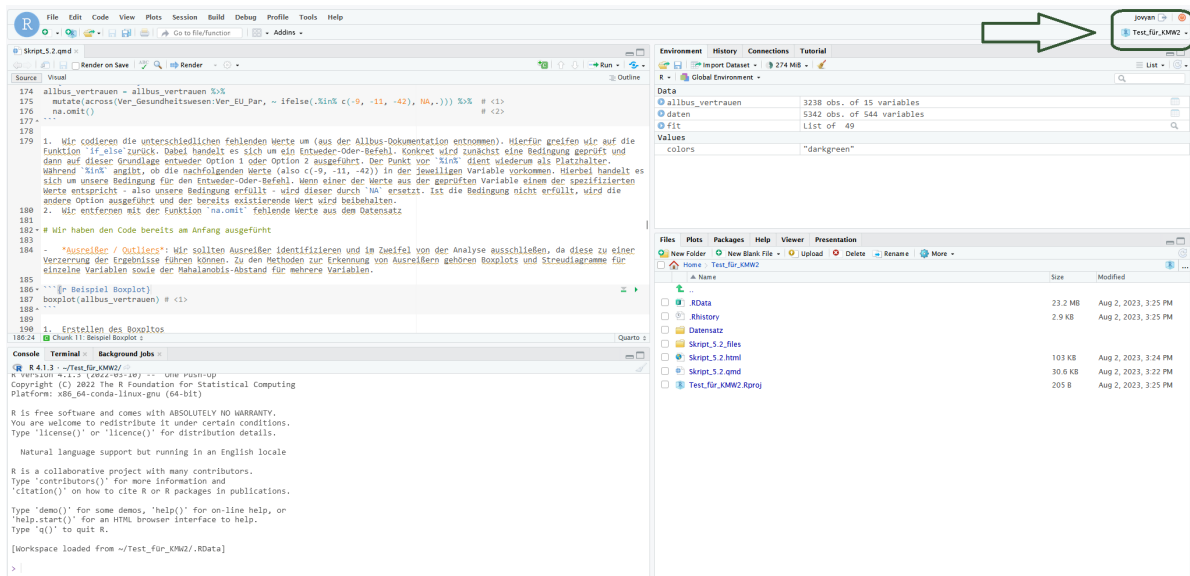


Figure 3: Screenshot RStudio

Nutzer:innen Interfaces auf den Button **New Project** und wählt dann **Version Control** aus.

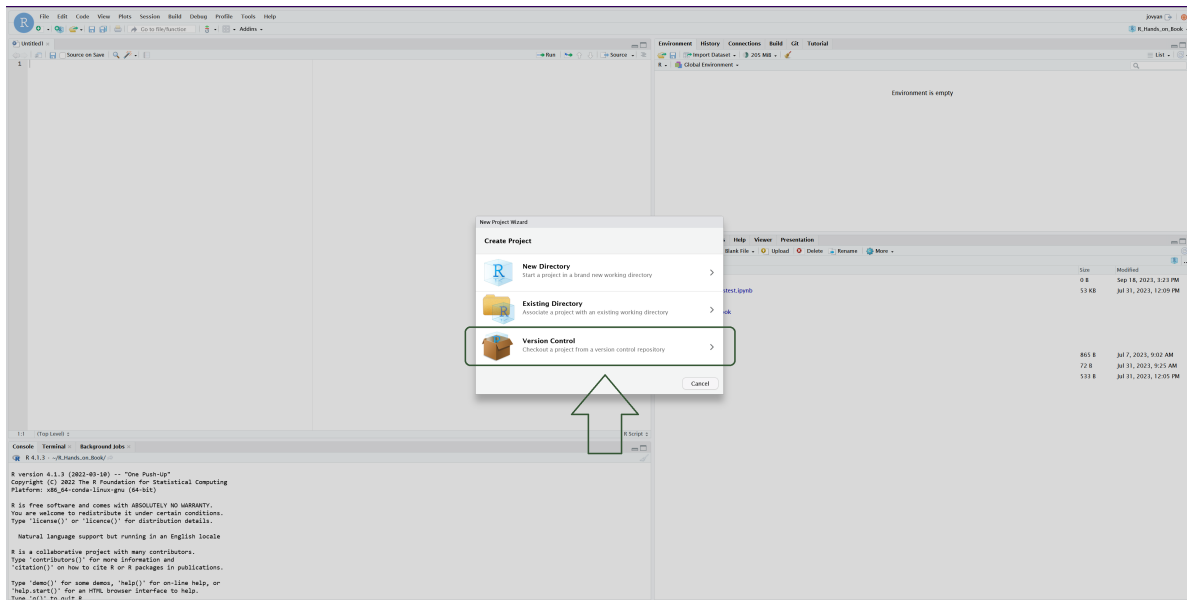


Figure 4: Screenshot RStudio

Im nächsten Schritt wählt ihr dann **Git** aus.

Und fügt unter *Repository URL* die folgende Adresse ein: https://github.com/patrickzerrer/R_Hands_on_Book.

Mit einem Klick auf den Button **Create Project** werden alle notwendigen Materialien und Ordner direkt in euer eigenes neugeschaffenes R-Projekt geladen.

Ihr habt jetzt alle Materialien und Skripte und könnt mit dem Kurs beginnen.

3 Das Working-Dictionary (wd)

Im Allgemeinen ist das Arbeitsverzeichnis (das Working-Dictionary, wd) der Ort, an dem R nach Dateien (vor allem nach Datensätzen) sucht. Wenn ihr kein Projekt verwendet, müsst ihr wahrscheinlich durch die Funktion `setwd` oder das Interface (siehe Screenshot) ein Working-Dictionary setzen, bevor ihr euren R-Code ausführen könnt.

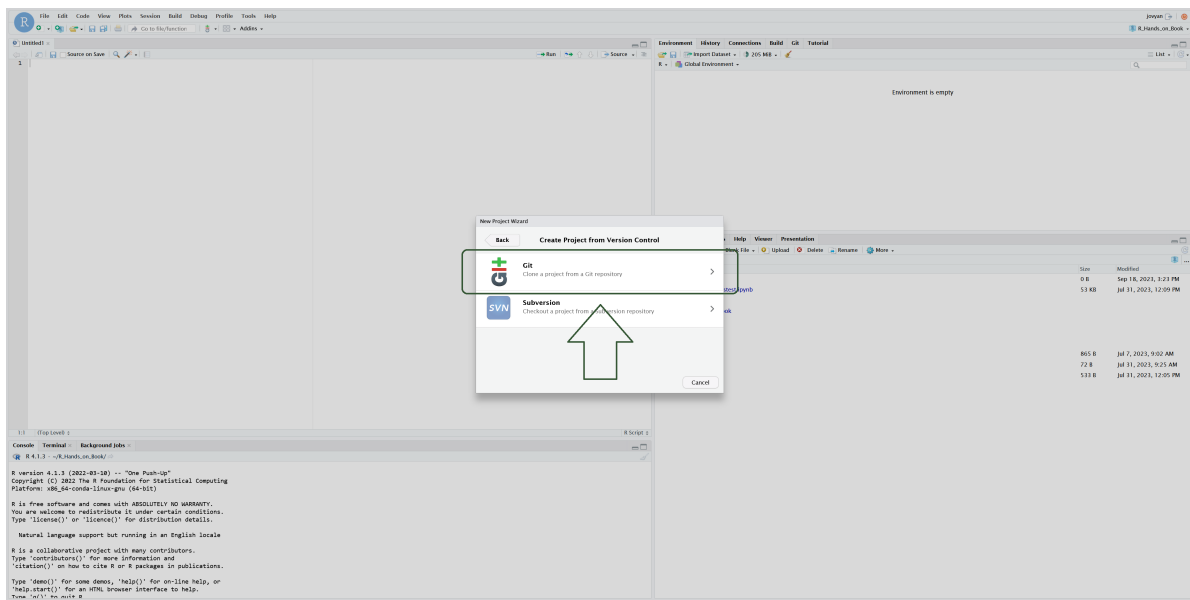


Figure 5: Screenshot RStudio

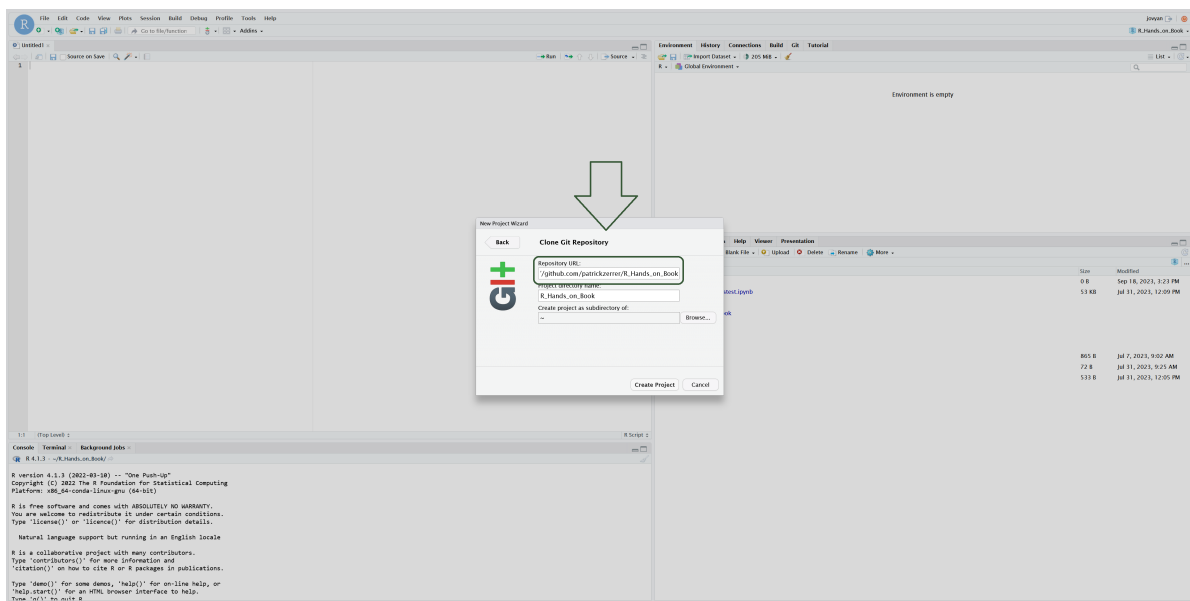
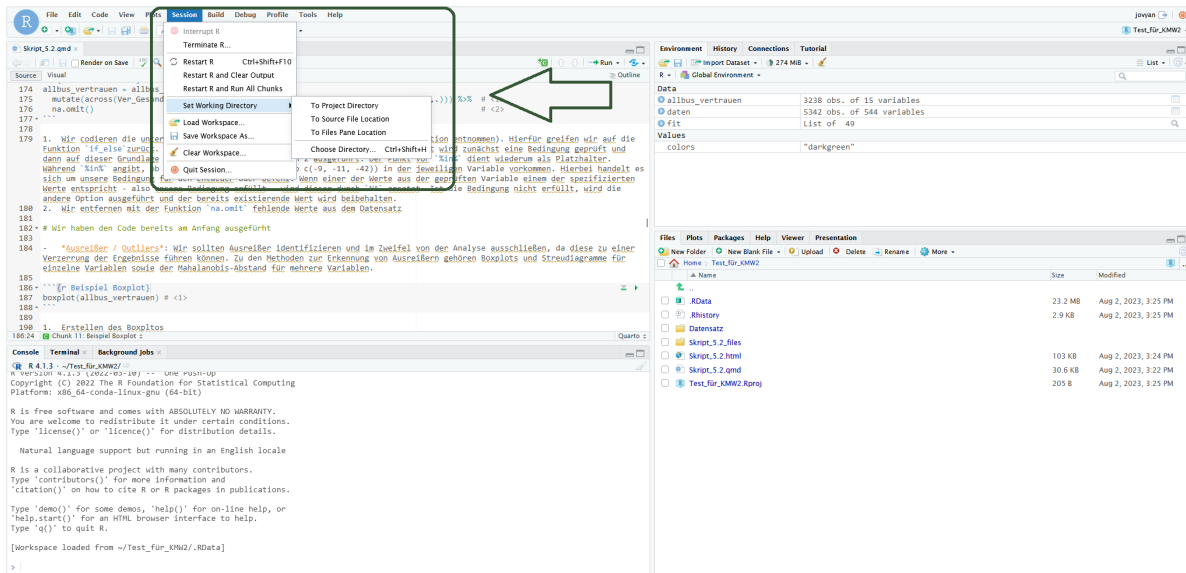


Figure 6: Screenshot RStudio



Wenn ihr beispielsweise den Code verwenden möchtet, den wir euch in diesem Kurs zu Verfügung stellen, müsst ihr darauf achten, dass ihr die Dateipfade auf euer Working-Dictionary und Ordnerstruktur anpasst.

`daten = haven::read_dta("Datensatz/Allbus_2021.dta")` ①

① Der Bereich in Anführungszeichen gibt den Dateipfad an. Damit sagt ihr R, wo sich die zu ladende Datei befindet.

In dem Beispiel ist im Working-Dictionary des Projekts ein Unterordner mit der zu ladenden Datei. Der Dateipfad befindet sich im R-Code in den Anführungszeichen, also "Datensatz/Allbus_2021.dta". Der erste Teil "Datensatz/" gibt an, dass sich die Datei in einem Ordner befindet, der Datensatz heißt. Der zweite Teil Allbus_2021.dta" gibt den eigentlichen Dateinamen mit der entsprechend Dateiendung an (also .dta). Wahrscheinlich müsst ihr den R-Code entsprechend anpassen.

4 Das Zugreifen auf Dateien und Datensätze

Bevor wir allerdings auf Dateien und Datensätze zugreifen können, müssen wir diese zunächst in unseren Projekt-Ordner (also in das Working-Dictionary) verschieben. Falls ihr für diesen Kurs die Desktop-Version von RStudio nutzt, könnt ihr hier einfach die entsprechenden Dateien auf eurem lokalen Rechner in das zuvor von euch festgelegte Working-Dictionary kopieren. Wenn ihr die Cloud-Version von RStudio nutzt, müsst ihr hier die Dateien und Datensätze zunächst in die Cloud laden. Hierfür könnt ihr auf den Upload-Button in der unteren rechten Ecke des Interfaces gehen und in dem sich öffnenden Fenster die entsprechende Datei auswählen.

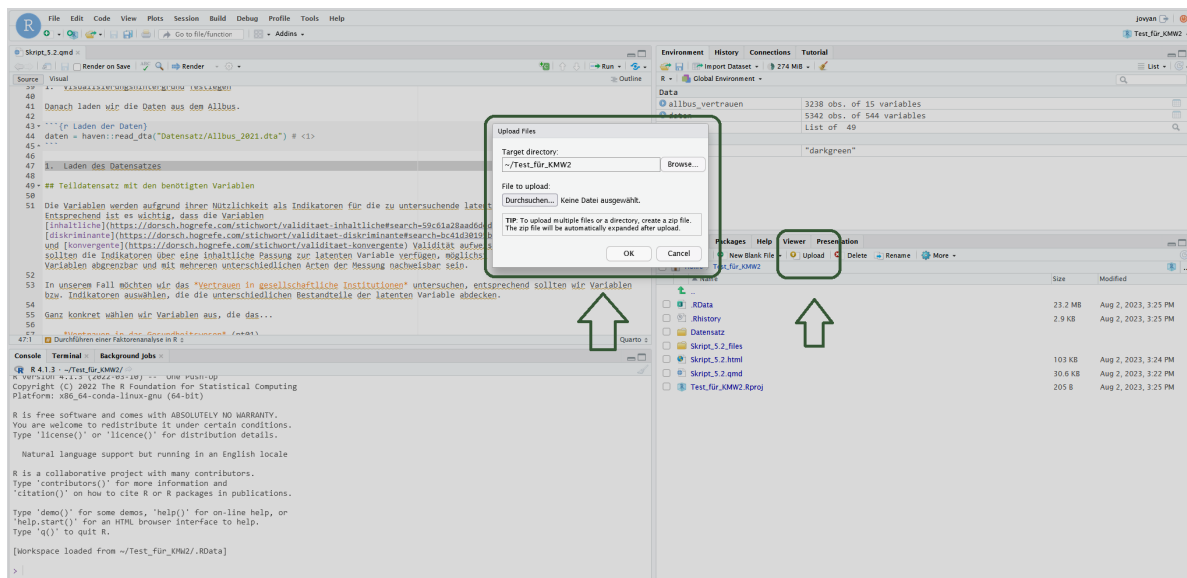


Figure 7: Screenshot RStudio

Sobald ihr durch das Klicken von OK bestätigt habt, wird die Datei hochgeladen und erscheint in dem unteren rechten Fenster im **Files**-Reiter.

5 Die Session

Die sogenannte *Session* wurde bereits mehrmals kurz erwähnt. Bei der *Session* handelt es sich um eine laufende Instanz des R-Programms, die ihr beenden und neustarten könnt. Das macht insbesondere dann Sinn, wenn ihr die falsche Version eines Pakets installiert habt oder ihr einen sauberen Start für euren R Code haben möchtet. Durch einen Neustart der *Session* könnt ihr euch sicher sein, dass ihr nicht ausversehen irgendwelche zuvor verwendeten Pakete oder Berechnungen mitnehmt.

Ihr könnt die *Session* mit der `.rs.restartR` Funktion in der Konsole neustarten. Alternativ könnt ihr im R Studio Interface auf **Session** und **Restart R** gehen.

Ein kurze Mitteilung in der Konsole gibt euch an, dass die *Session* neu gestartet wurde.

6 Hilfe zur Selbsthilfe

Ihr werdet im Laufe des Kurses immer wieder mit Fehlermeldungen zu tun haben. Hier ist es wichtig einen kühlen Kopf zu bewahren und möglichst systematisch zu versuchen den Fehler zu

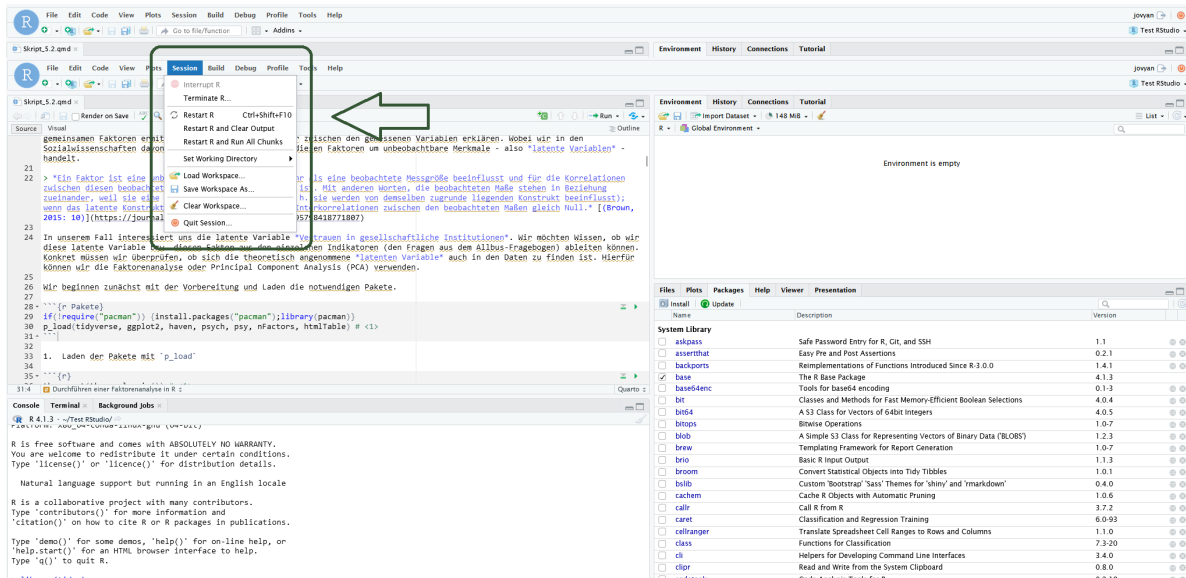


Figure 8: Screenshot RStudio

finden und zu beheben. Der Umgang mit Fehlern ist ein großer und wichtiger Teil bei jeglicher Programmierung und es gibt hier ein paar Tipps, wie man damit am besten umgeht.

1. *Atmen:* Bleib ruhig und atme ein paar mal tief durch, gerade wenn es nicht der erste Fehler ist der dir heute begegnet.
2. *Lesen:* Lies aufmerksam die Fehlermeldung.
3. *Code überprüfen:* Schaue nach, ob sich ein Warnzeichen am Rand deines R Codes befindet. Dieses Warnzeichen weist dich auf einen Syntaxfehler (bspw. eine vergessene Klammer oder Komma) hin.
4. *Variablen überprüfen:* Überprüfe kurz, ob du die richtigen Variablen verwendest oder sich ein Tippfehler eingeschlichen hat. Achtet auch auf Groß- und Kleinschreibung.
5. *Informationen suchen:* Rufe eine Vignette (mit dem Befehl `vignette`) oder rufe die Hilfeseite mit einem der Funktion vorgelagerten Fragezeichen auf (mit bspw. `?dplyr()`) auf und lies sie dir durch.
6. *Sich im Netz Hilfe suchen:* Suche auf [Stack Overflow](https://stackoverflow.com) nach deinem Problem, meistens hat jemand auf der Welt das oder ein ähnliches Problem schon einmal gelöst.
7. *Frag deine Kommilitonen:* Tauscht euch aus, gemeinsam findet ihr eine Lösung!
8. *Die KI fragen:* Nutze ein KI Chatbot, um dir die Fehlermeldung erklären zu lassen und Lösungsvorschläge zu unterbreiten. Beachte hierbei bitte, dass die Antworten nicht

immer korrekt sind und meist Kontextwissen von dir benötigt wird, um die Korrektheit der Antworten zu prüfen [Link](#).

9. *Frag uns*: Frag dein:e Dozent:in :)