Model

Datensatz

Ab dieser Woche verwenden wir einen neuen Datensatz, welcher verschiedene demografische und schulrelevante Daten von Schüler*innen aus Portugal erfasst. Es handelt sich dabei um echte Daten, die von Paulo Cortez erhoben wurden und Kaggle herunter geladen werden können. Ingsesamt wurden die Daten von 395 Schüler*innen erhoben. Der Artikel zum Datensatz kann hier gefunden werden. In den nächsten Wochen untersuchen wir, wodurch die Mathematikleistung dieser Schüler beeinflusst wird.

Der Datensatz hat den Namen `student_data` und kann hier herunter geladen werden. Die Variablen des Datensatzes sind:

1. school - student's school (binary: 'GP' - Gabriel Pereira or 'MS' - Mousinho da Silveira)
2. sex - student's sex (binary: 'F' - female or 'M' - male)
3. age - student's age (numeric: from 15 to 22)
4. address - student's home address type (binary: 'U' - urban or 'R' - rural)
5. famsize - family size (binary: 'LE3' - less or equal to 3 or 'GT3' - greater than 3)
6. Pstatus - parent's cohabitation status (binary: 'T' - living together or 'A' - apart)
7. Medu - mother's education (numeric: 0 - none, 1 - primary education (4th grade), 2 – 5th to 9th grade, 3 – secondary education or 4 – higher education)
8. Fedu - father's education (numeric: 0 - none, 1 - primary education (4th grade), 2 – 5th to 9th grade, 3 – secondary education or 4 – higher education)
9. Mjob - mother's job (nominal: 'teacher', 'health' care related, civil 'services' (e.g. administrative or police), 'at_home' or 'other')
10. Fjob - father's job (nominal: 'teacher', 'health' care related, civil 'services' (e.g. administrative or police), 'at_home' or 'other')
11. reason - reason to choose this school (nominal: close to 'home', school 'reputation', 'course' preference or 'other')
12. guardian - student's guardian (nominal: 'mother', 'father' or 'other')
13. traveltime - home to school travel time (numeric: 1 - <15 min., 2 - 15 to 30 min., 3 - 30 min. to 1 hour, or 4 - >1 hour)
14. studytime - weekly study time (numeric: 1 - <2 hours, 2 - 2 to 5 hours, 3 - 5 to 10 hours, or 4 - >10 hours)
15. failures - number of past class failures (numeric: n if 1<=n<3, else 4)
16. schoolsup - extra educational support (binary: yes or no)
17. famsup - family educational support (binary: yes or no)
18. paid - extra paid classes within the course subject (Math or Portuguese) (binary: yes or no)

19. activities - extra-curricular activities (binary: yes or no)
20. nursery - attended nursery school (binary: yes or no)
21. higher - wants to take higher education (binary: yes or no)
22. internet - Internet access at home (binary: yes or no)
23. romantic - with a romantic relationship (binary: yes or no)
24. famrel - quality of family relationships (numeric: from 1 - very bad to 5 - excellent)
25. freetime - free time after school (numeric: from 1 - very low to 5 - very high)
26. goout - going out with friends (numeric: from 1 - very low to 5 - very high)
27. Dalc - workday alcohol consumption (numeric: from 1 - very low to 5 - very high)
28. Walc - weekend alcohol consumption (numeric: from 1 - very low to 5 - very high)
29. health - current health status (numeric: from 1 - very bad to 5 - very good)
30. absences - number of school absences (numeric: from 0 to 93)
31. G1 - first period grade mathematics (numeric: from 0 to 20)
32. G2 - second period grade mathematics (numeric: from 0 to 20)
33. G3 - final grade mathematics (numeric: from 0 to 20, output target)

In dieser Woche werden wir uns im besonderen mit den Variablen G3, failures und studytime beschäftigen.

Das Modell

Im letzten Modul haben wir uns das zwei-parameter Modell angeschaut, bei dem sowohl die abhängige als auch die unabhängige Variable intervallskaliert war:

$$Y_i = \beta_0 + \beta_1 * X_{i1} + \epsilon_i$$

Beispielsweise könnten wir auf Grundlage des einfachen Regressionsmodells fragen, ob die Lernzeit von Schülern einen Einfluss auf deren Mathematikleistung hat. Eine andere Fragestellung wäre, ob die Anzahl der Fächer in denen ein*e Schüler*in durchgefallen ist,einen Einfluss auf die Mathematikleistung hat? Um beide Fragestellungen zu beantworten, könnten wir zwei einfache Regressionsmodelle berechnen. Wir werden allerdings in diesem Modul lernen, beide Fragestellungen anhand der multiplen Regression zu testen, indem wir mehrere Prädiktoren in unser Modell hinzunehmen:

$$Y_i = \beta_0 + \beta_1 * X_{i1} + \beta_2 * X_{i2} + ... + \beta_p * X_{i,p-1} + \epsilon_i$$

Y_i steht erneut für unsere abhängige Variable und ϵ_i steht für die Fehler, welche unser Modell nicht erklären kann. X_{ij} steht für die Werte unserer intervallskalierten abhängigen Variablen, beispielsweise die Dauer der Lernzeit einer bestimmten Schülerin. β_j steht für die **partiellen Regressionskoeffizienten**. Wir werden später ausführlich darüber reden, weshalb diese Koeffizienten partiell heißen. Fur jetzt genügt es zu wissen, dass diese partiellen Regressionskoeffizienten von den anderen Prädiktoren abhängig sind und sich mit dem Entfernen bzw. Hinzufügen dieser Prädiktoren ändern.

In diesem Modul werden wir eine multiple Regression mit zwei Prädiktoren berechnen. Das zugehörige Modell sieht folgendermaßen aus:

$$Y_i = \beta_0 + \beta_1 * X_{i1} + \beta_2 * X_{i2} + \epsilon_i$$

Folgende Variablen werden wir in dem Modell untersuchen:

- Y_i (abhängige Variable) - `G3`: Die Mathematikleistung der SuS (0 bis 20 Punkte)
- X_{i1} (unabhängige Variable) - `failures`: Die Anzahl der Fächer, indenen der/die Schüler*in durchgefallen
- X_{i2} (unabhängige Variable) - `studytime`: Die wöchentliche Zeit, die die SuS auf das Lernen der Mathematik aufwenden

Während wir uns in der einfachen linearen Regression das Model als eine Linie vorstellen können, können wir uns dieses multiple lineare Regressionsmodell als eine Fläche darstellen.

Bestimmung der Betagewichte

> Die Berechnung der Betagewichte durch Verfahren der linearen Algebra ist nicht prüfungsrelevant

Die Berechnung der Betagewichte ist bei der multiplen Regression deutlich schwieriger als bei der einfachen linearen Regression. Konzeptuell versuchen wir erneut die Betagewichte zu finden, die die quadrierten Abweichungen der realen Werte von den vorhergesagten Werte minimieren:

$$\min \sum_{i=1}^{N} (Y_i - \hat{Y}_i)^2$$

Wir können diese Betagewichte erneut durch lineare Algebra berechnen:

$$(A^T A)^{-1} A^T b$$

A steht für eine Matrix der unabhängigen Variablen (X_1). b steht für Y.

Zunächst verfassen wir A:

```r
rows <- nrow(student_mat)

A <- matrix(
  c(rep(1, rows),
  student_mat$failures,
  student_mat$studytime),
  nrow = rows)
b <- matrix(student_mat$G3, nrow = rows)
solve(t(A) %*% A) %*% t(A) %*% b
```

```
           [,1]
[1,] 10.7401929
[2,] -2.1815446
[3,]  0.1984922
```

Diese Betagewichte können wir mit Hilfe der Funktion lm prüfen:

```r
lm(G3 ~ failures + studytime, data = student_data)
```

```

Call:
lm(formula = G3 ~ failures + studytime, data = student_mat)

Coefficients:
(Intercept)     failures    studytime
    10.7402      -2.1815       0.1985
```

Unser Modell lautet daher:

$$\begin{aligned}\hat{Y}_i &= b_0 + b * X_{i1} + b * X_{i2} + e_i \\ &= 10.74 + (-2.18) * X_{i1} + 0.19 * X_{i2} + e_i \\ &= 10.74 - 2.18 * X_{i1} + 0.19 * X_{i2} + e_i\end{aligned}$$

Partielle Regressionskoeffizienten

Interpretation der Prädiktoren

Unser multiples Regressionsmodell sieht nun folgendermaßen aus:

$$\hat{Y}_i = 10.74 - 2.18 * X_{i1} + 0.19 * X_{i2} + e_i$$

Die naheliegende Frage ist, was die Regressionskoeffizienten besagen? Zunächst können wir auf Grundlage des Modells $\hat{Y}$ berechnen. Stellen wir uns eine Schülerin vor, die 5 bis 10 Stunden pro Woche lernt und daher für `studytime` den Wert 3 erhält und die bisher einmal in einem Fach durchgefallen ist:

$$\begin{aligned}\hat{Y}_i &= 10.74 - 2.18 * X_{i1} + 0.19 * X_{i2} + e_i \\ &= 10.74 + (-2.18) * 1 + 0.19 * 3 + e_i \\ &= 9.13 + e_i\end{aligned}$$

$\hat{Y}$ wäre in diesem Fall 0.13. Wir schätzen, dass die Schülerin 9.13 von 20 möglichen Punkten in der Klausur enthält.

Wir können die Regressionskoeffizienten zudem einzeln interpretieren. Der Koeffizient der Variable `failures` ist -2.18. Dies bedeutet, dass die Mathematiknote mit jedem Durchfallen in einem Fach um 2.8 Punkte absinkt. Schüler, die daher noch nie durchgefallen sind, sollten eine bessere Note erhalten, als Schüler, die bereits ein- oder mehrmals durchgefallen sind. Der Koeffizient der Variable `studytime` besagt, dass mit jeder weiteren Stufe der Lernzeit, die Mathematikleistung um 0.19 Punkte zunimmt.

Beide Koeffizienten allerdings sind nur in **Abhängigkeit** des anderen Prädiktors zu interpretieren. Wir würden sagen, dass wir für andere Prädiktoren **kontrollieren**. Daher sprechen wir von **partiellen Regressionskoeffizienten**.

Redundanz der Prädiktoren

Sobald Prädiktoren miteinander korrelieren, herrscht eine gewissen Redundanz zwischen diesen. Stell dir ein extremes Beispiel vor. Du möchtest das Gewicht einer Person auf Grundlage der Größe einer Person in Zentimeter und der Größe einer Person in Inch berechnen. Die Größe einer Person in Zentimeter und Inch ist komplett redundant, was sich dadurch zeigen lässt, dass deren Korrelation 1 ist. Der zweite Prädiktor wird daher so gut wie keine weiteren Fehler im Vergleich zum ersten Prädiktor erklären und daher komplett redundant sein.

Meistens sind Prädiktoren nicht so hoch miteinander korreliert. In unserem Beispiel beläuft sich die Korrelation beider Variablen auf -0.17:

```
cor(student_mat$failures, student_mat$studytime) # -0.17356
```

Die Folge ist, dass sich die Betagewichte zwischen der einfachen Regression und der multiplen Regression unterscheiden. Hier siehst die Betagewichte mit der Variable `studytime` als unabhängige Variable bei einer einfachen Regression:

$$\hat{Y}_i = 9.328 + 0.534 * X_{i1} + e_i$$

b_1 ändert sich allerdings, wenn wir den zweiten Prädiktor hinzunehmen:

$$\hat{Y}_i = 10.74 - 2.18 * X_{i1} + 0.19 * X_{i2} + e_i$$

Während b_1 bei der einfachen Regression 0.53 beträgt, beläuft sich b_1 bei der multiplen Regression auf 0.19. Dies liegt an der Redundanz der beiden Prädiktoren geschuldet. Wären beide Prädiktoren gar nicht miteinander korreliert, wären diesen beiden Regressionskoeffizienten identisch.

Im nächsten Schritt versuchen wir zu verstehen, weshalb dieser Unterschied besteht und wie dieser zu Stand kommt.

Berechnung partieller Regressionskoeffizienten

Das Ziel diesen Abschnitts ist es, die partiellen Regressionskoeffizienten zu berechnen und zu verstehen. Hierfür vergegenwärtigen wir uns erneut unser berechnetes Modell:

$$\hat{Y}_i = 10.74 - 2.18 * X_{i1} + 0.19 * X_{i2} + e_i$$

Wir hatten gesagt, dass die Werte -2.18 und 0.19 partielle Regressionskoeffizienten sind. Das bedeutet, dass diese in *Abhängigkeit* des jeweilig anderen Prädiktors stehen bzw. für den anderen Prädiktor kontrolliert sind.

Wir werden im nächsten Schritt mehrere Modelle berechnen, die jeweils aufeinander aufbauen. Im letzen einfachen Regressionsmodell werden wir den gleichen Regressionskoeffizienten erhalten wie in der multiplen Regression, mit dem Unterschied, dass wir diesen besser intepretieren können. Die Darstellung dieser Modelle dient dir dazu, die partiellen Regressionskoeffizienten intuitiv zu verstehen.

Dieser Teil ist inhaltlich komplex, nehm dir daher genug Zeit, ihn zu verstehen.

Model 1: Einfaches Model mit dem Mittelwert der Mathematikleistung

Zunächst erstellen wir uns Modell mit einem Parameter, dem Mittelwert der abhängigen Variable:

$$\begin{aligned} Y_{math} &= b_0 + e_i \\ Y_{math} &= 10.42 + e_i \end{aligned}$$

Wir können die Gleichung umstellen und zeigen, dass der Fehler durch $e_i = Y_{math} - 10.42$ berechnet werden kann. Speichern wir diesen Fehler in einer eigenen Variable:

```
(residual_dataframe <- student_data %>%
  rownames_to_column(var = "id") %>%
  select(id, G3, studytime, failures) %>%
```

```
  mutate(
    e_y_math = G3 - mean(G3)
  ))

# A tibble: 395 x 5
   id       G3 studytime failures e_y_math
   <chr> <dbl>     <dbl>    <dbl>    <dbl>
 1 1         6         2        0   -4.42
 2 2         6         2        0   -4.42
 3 3        10         2        3   -0.415
 4 4        15         3        0    4.58
 5 5        10         2        0   -0.415
 6 6        15         2        0    4.58
 7 7        11         2        0    0.585
 8 8         6         2        0   -4.42
 9 9        19         2        0    8.58
10 10       15         2        0    4.58
# ... with 385 more rows
```

Der Schüler mit der ID 2 beispielsweise, erhält in der Mathematikleistung in Wirklichkeit die Punktzahl 6, wir überschätzen diese Punktzahl allerdings um 4.42 Punkte. Negative Werte bedeuten daher, dass wir die Punktzahl auf Grundlage des einfachen Modells überschätzen, negative Werte bedeuten, dass wir die Punktzahl unterschätzen.

Model 2: Einfaches Model mit dem Mittelwert der Durchfallrate

Ein ähnliches Modell stellen wir im nächsten Schritt für die Variable `failures` auf:

$$Y_{failures} = b_0 + e_i$$
$$Y_{failures} = 0.33 + e_i$$

Erneut erhalten wir für die Fehler $e_i = Y_{failures} - 0.33$ und können diese Daten in unseren Datensatz einfügen:

```
(residual_dataframe <- residual_dataframe %>%
  mutate(
    e_y_failures = failures - mean(failures)
  ))

# A tibble: 395 x 6
   id       G3 studytime failures e_y_math e_y_failures
   <chr> <dbl>     <dbl>    <dbl>    <dbl>        <dbl>
 1 1         6         2        0   -4.42        -0.334
 2 2         6         2        0   -4.42        -0.334
```

```
 3 3         10          2         3   -0.415         2.67
 4 4         15          3         0    4.58         -0.334
 5 5         10          2         0   -0.415        -0.334
 6 6         15          2         0    4.58         -0.334
 7 7         11          2         0    0.585        -0.334
 8 8          6          2         0   -4.42         -0.334
 9 9         19          2         0    8.58         -0.334
10 10        15          2         0    4.58         -0.334
# ... with 385 more rows
```

Unser Schüler mit der ID 2 ist in Wirklichkeit noch nicht durchgefallen, wir schätzen allerdings seine Durchfallquote auf 0.33 und überschätzen daher seine Durchfallquote.

Model 3: e_{y-math} auf Grundlage von $e_{y-failures}$ regredieren

Auf Grundlage dieser beiden Fehlerquellen können wir uns nun fragen, ob ein Schüler, der öfter als gewöhnlich durch ein Fach fliegt auch schlechter als der Durschschnitt in der Mathematik ist? Da der Mittelwert von e_{y-math} und $e_{y-failures}$ 0 beträgt, können wir auf den Intercept verzichten:

$$e_{y-math} = \beta_1 * e_{y-failures}$$

Eine lineare Regression dieser beiden Variablen ergibt:

$$e_{y-math} = -2.22 * e_{y-failures}$$

Diese Modell können wir wie folgt interpretieren: Schüler, die öfters als der Durchschnitt durch einen Kurs fliegen, sind ebenso schlechter in der Mathematikleistung als der Durschschnitt. Wäre b_1 positiv, würden wir davon ausgehen, dass Schüler die öfter als der Durschschnitt durch einen Kurs fliegen, **besser** in der Mathematikleistung sind als der Durchschnitt.

Bei unserem Schüler mit der ID zwei würden wir daher davon ausgehen, dass er 0.07 Punkte besser in der Mathematiknote als der Durchschnitt ist, da er im Schnitt weniger durch die Kurse fällt als seine Klassenkameraden:

$$e_{y-math} = -2.22 * (-0.334) = 0.07$$

Erneut können wir die Fehler dieses Modells berechnen, indem wir die geschätze Abweichung der Mathematiknote vom Mittelwert von der tatsächlichen Abweichung vom Mittelwert berechnen:

```r
(residual_dataframe <- residual_dataframe %>%
  mutate(
    e_math_failures = e_y_math - (-2.22 * e_y_failures)
  ))
```

```
# A tibble: 395 x 7
   id       G3 studytime failures e_y_math e_y_failures e_math_failures
   <chr> <dbl>     <dbl>    <dbl>    <dbl>        <dbl>           <dbl>
 1 1         6         2        0   -4.42        -0.334          -4.49
 2 2         6         2        0   -4.42        -0.334          -4.49
 3 3        10         2        3   -0.415        2.67            0.171
 4 4        15         3        0    4.58        -0.334           4.51
 5 5        10         2        0   -0.415       -0.334          -0.489
 6 6        15         2        0    4.58        -0.334           4.51
 7 7        11         2        0    0.585       -0.334           0.511
 8 8         6         2        0   -4.42        -0.334          -4.49
 9 9        19         2        0    8.58        -0.334           8.51
10 10       15         2        0    4.58        -0.334           4.51
# ... with 385 more rows
```

Wir können `e_math_failures` nun folgendermaßen interpretieren. Kontrolliert für die Anzahl der Male, die eine Person durch einen Kurs gefallen ist, ist Person X `e_math_failures` besser oder schlechter als der Durchschnitt.

Model 4: Die Lernzeit auf Grundlage der Anzahl der Durchfallquote regredieren

Um zu überprüfen, weshalb nun eine Person eine schlechtere Mathematikleistung als der Durschnitt erhält, wenn wir für die Anzahl der Male, die eine Person durchgefallen ist kontrollieren, können wir die Variable `studytime` hinzunehmen. Diese ist allerdings mit der Variable `failures` redundant. Aus diesem Grund müssen wir den Anteil der Lernzeit berechnen, die nicht redundant zu der Male ist, die eine Person durchgefallen ist. Dies können wir schaffen, indem wir die Fehler berchnen, die entstehen, wenn wir die Lernzeit auf Grundlage der Durchfallquote der Personen regredieren:

$$Y_{studytime} = b_0 + b_1 * X_{i-failures} + e$$

Hieraus ergibt sich:

$$\hat{Y}_{studytime} = 2.1009 - 0.1959 * X_{i-failures}$$

Dieses Modell sagt nun voraus, dass mit jedem Mal, das eine Person durchfällt, die Lernzeit um 0.19 Punkte absinkt. Unsere Person mit der ID beispielsweise hat eine Lernzeit von 2, wir würden allerdings annehmen, dass diese $2.1 - 0.19 * 0 = 2.1$

beträgt. Der Fehler beläuft sich daher auf $2.0 - 2.1 = -0.1$. Berechnen wir diese Fehler für alle SuS:

```
(residual_dataframe <- residual_dataframe %>%
  mutate(
    e_study_failures = studytime - (2.1009 - 0.1959 * failures)
  ))
```

```
# A tibble: 395 x 8
   id       G3 studytime failures e_y_math e_y_failures e_math_failures e_study_failures
   <chr> <dbl>     <dbl>    <dbl>    <dbl>        <dbl>           <dbl>            <dbl>
 1 1         6         2        0   -4.42        -0.334          -5.16            -0.101
 2 2         6         2        0   -4.42        -0.334          -5.16            -0.101
 3 3        10         2        3   -0.415        2.67            5.50             0.487
 4 4        15         3        0    4.58        -0.334           3.84             0.899
 5 5        10         2        0   -0.415       -0.334          -1.16            -0.101
 6 6        15         2        0    4.58        -0.334           3.84            -0.101
 7 7        11         2        0    0.585       -0.334          -0.157           -0.101
 8 8         6         2        0   -4.42        -0.334          -5.16            -0.101
 9 9        19         2        0    8.58        -0.334           7.84            -0.101
10 10       15         2        0    4.58        -0.334           3.84            -0.101
# ... with 385 more rows
```

Model 5: Kontrollierte Mathematikleistung auf Grundlage der kontrollierten Lernzeit regredieren

Zuletzt können wir eine einfache Regression aufstellen, in der wir die durchschittliche Mathematikleistung kontrolliert für die Anzahl der Male, die eine Person durchgefallen ist anhand der durchschnittlichen Lernzeit ebenso kontrolliert für die Anzahl der Male die eine Person durchgefallen ist, regredieren:

$$e_{math-failures} = b_0 * e_{study-failures}$$

Dies ergibt:

$$e_{math-failures} = 0.1985 * e_{study-failures}$$

Verglichen mit unserer multiplen Regression erhalten wir daher den gleichen Regressionskoeffizienten:

```
lm(G3 ~ studytime + failures, data = student_mat)
```

```
Call:
lm(formula = G3 ~ studytime + failures, data = student_mat)

Coefficients:
```

```
(Intercept)    studytime     failures
    10.7402       0.1985      -2.1815
```

Wir können daher den Regressionskoeffizienten b_1 folgendermaßen interpretieren:

> Für jede Einheit der Variable `studytime`, die eine Person mehr studiert als für die Anzahl der Male, die die Person durchgefallen ist, zu erwarten ist, schätzen wir, dass die Person 0.1985 Punkte besser in der Mathematik abschneidet als man für die Male erwarten würde, die diese Person durchgefallen ist.

Dieser Satz ist komplex und vermutlich brauch es eine Weile, ihn zu verstehen. Wichtig ist, dass du dir vergegenwärtigst, dass die Regressionskoeffizienten immer in Abhängigkeit der anderen Prädiktoren zu interpretieren sind. Je stärker die Prädiktoren miteinander korrelieren, desto stärker ist diese Abhängigkeit.

Statistische Inferenz

Das gesamte Modell

Um die einzelnen Parameter (b_0, b_1 und b_1) in unserer multiplen Regression zu testen, können wir das gleiche Verfahren verwenden, welches wir bereits in den anderen Modulen kennen gelernt haben. (1) Zunächst stellen wir unser erweitertes und kompaktes Modell auf. (2) Anschließend berechnen wir das F für beide Modelle. (3) Zuletzt berechnen wir die Wahrscheinlichkeit für das F unter Annahme der Nullhypothese. (4) Zum Schluss berechnen wir die Effektgröße und berichten unser Ergebnis. Erneut möchten wir prüfen, ob das Hinzufügen von Parametern die Fehler so stark reduziert, dass wir rechtfertigen können, diesen Parameter in das Modell aufzunehmen.

Beginnen wir mit einem Test, welcher folgende beiden Modelle miteinander vergleicht:

$$MODEL\ A = Y_i = \beta_0 + \beta_1 * X_1 + \beta_2 * X_2 + \epsilon_i$$
$$MODEL\ C = Y_i = B_0 + \epsilon_i$$

Wir testen demnach, ob die beiden weiteren Parameter Lernzeit und Anzahl der Male, die eine Person durchgefallen ist, die Fehler des einfachen Modells des Mittelwerts substantiell reduziert. Wir gehen daher bei der Nullyhypothese davon aus, dass die weiteren Parameter die abhängige Variable nicht verändern und daher 0 sind:

$$H_0 : \beta_0 = \beta_1 = \beta_2 = 0$$

Im nächsten Schritt berechnen wir F analog, wie wir es bisher gemacht haben:

```r
mean_sample <- mean(student_data$G3) # Mittelwert der Stichprobe
errors <- student_data %>%
  mutate(
    compact_model = mean_sample,
    augmented_model = 10.7402 + 0.1985 * studytime - 2.1815 * failures,
    res_compact   = (G3 - compact_model)**2,
    res_augmented = (G3 - augmented_model)**2
  )

(sse_c <- sum(errors$res_compact)) # 8269.909
(sse_a <- sum(errors$res_augmented)) # 7185.053
(ssr <- sse_c - sse_a) # 1084.856

(pre <- ssr / sse_c) # 0.1311812
```

Der F-Wert ist daher:

$$F = \frac{SSR/(PA - PC)}{SSE(A)/(n - PA)}$$

- PC: Das kompakte Modell hat einen Parameter B_0.
- PA: Das erweiterte Modell hat drei Parameter: b_0, b_1 und b_2.
- n: Insgesamt gibt es 395 Personen in dem Datensatz.

```r
(F <- (ssr / (3 - 1)) / (sse_a / (395 - 3))) # 29.59364
```

Die Wahrscheinlichkeit für einen solch hohen F-Wert ist deutlich unter dem kritischen Wert:

```r
1 - pf(F, df1 = 2, df2 = 395 - 3) #  1.071809e-12
```

Unsere übliche Tabelle lautet daher:

Source	SS	df	MS	F	p	PRE / R^2
Reduction	1084.856	2	542.43	29.59	$< .001$	0.13
Error	7185.053	392	18.33			
Total Error	8269.909	394				

Probleme des allgemeinen Modells

Ein solcher Test ist allerdings nicht sonderlich hilfreich, da wir auf Grundlage des Ergebnisses nicht hervorsagen können, inwieweit die **einzelnen** Parameter die Fehler substantiell reduzieren. Auf Grundlage des Ergebnisses können wir lediglich sagen, dass das Hinzufügen **beider** Parameter, den Fehler substantiell reduziert. Wir wissen jedoch nicht, ob beide oder nur einer dieser Parameter für diese Fehlerreduktion zuständig ist?

Als Faustregel: Sobald der Freiheitsgrad des Zählers über 1 ist, können wir das Ergebnis nur schwer interpretieren, da mehrere Parameter für die Reduktion des Fehlers ausschlagbend sein können. Wir müssen daher einen Weg finden, den Freiheitsgrad auf 1 zu setzen, um eine Interpretation zu ermöglichen und das erweiterte und kompakte Modell so einzugrenzen, dass sich diese nur in einem Parameter unterscheiden.

Einzelne Prädiktoren testen

Parameter Studytime

Um nun den Beitrag der einzelnen Parameter zu testen, müssen wir alternative Modelle gegeneinander testen. Beginnen wir, indem wir den Beitrag des Parameters `studytime` testen:

$$MODEL\ A = \beta_0 + \beta_{failures} * X_{failures} + \beta_{studytime} * X_{studytime} + \epsilon_i$$
$$MODEL\ C = \beta_0 + \beta_{failures} * X_{failures} + \epsilon_i$$

Du erkennst, dass sich beide Modelle nur in einem Parameter unterschieden. Das kompakte Modell hat zwei Parameter, das erweiterte Modell drei Parameter. Folgende Modelle ergeben sich hieraus:

$$MODEL\ A = 10.7402 - 2.1815 * X_{failures} + 0.1985 * X_{studytime}$$
$$MODEL\ C = 11.16 - 2.22 * X_{failures}$$

Der F-Wert ist folgerichtig:

```
mean_sample <- mean(student_data$G3)
errors <- student_data %>%
  mutate(
    compact_model   = 11.16 - 2.22 * failures,
    augmented_model = 10.7402 - 2.1815 * failures + 0.1985 * studytime,
    res_compact     = (G3 - compact_model)**2,
    res_augmented   = (G3 - augmented_model)**2
  )

(sse_c <- sum(errors$res_compact)) # 7195.66
(sse_a <- sum(errors$res_augmented)) # 7185.053
(ssr <- sse_c - sse_a) # 10.6075

(pre <- ssr / sse_c) # 0.001474152

(F <- (ssr / (2 - 1)) / (sse_a / (395 - 2))) # 0.580197

1 - pf(F, df1 = 1, df2 = 395 - 2) # 0.4466919
```

Source	SS	df	MS	F	p	PRE / R^2
studytime	10.6075	1	10.6075	0.58	0.447	0.00
Error	7185.053	393	18.28258			
Total Error	7195.66	394				

Wir können auf Grundlage dieses Ergebnisses sagen, dass der Parameter `studytime` den Fehler nicht substantiell reduziert und daher nicht signifikant ist. Der Parameter reduziert den Fehler nicht mehr als ein willkürlicher Parameter, den wir einfach so ein das Modell hinzunehmen.

Manche Statistikprogramme berichten anstatt des F-Wertes den t-Wert. Wir wissen allerdings mittlerweile, dass der t-Wert nichts anderes ist als die Wurzel des F-Wertes. Wir könnten anstatt F daher t als $\sqrt{0.58} = 0.761$ berichten.

Parameter Failures

Das gleiche können wir für den Parameter `failures` berechnen: Führt das Hinzufügen des Parameters `failures` zu einer substantiellen Reduzierung des Fehlers im Vergleich zu dem Regressionsmodells, welches diesen Parameter nicht besitzt?

$$MODEL\ A = 10.7402 - 2.1815 * X_{failures} + 0.1985 * X_{studytime}$$
$$MODEL\ C = 9.328 + 0.534 * X_{studytime}$$

Hieraus ergibt sich:

```
mean_sample <- mean(student_data$G3)
errors <- student_data %>%
  mutate(
    compact_model   = 9.328 + 0.534 * studytime,
    augmented_model = 10.7402 - 2.1815 * failures + 0.1985 * studytime,
    res_compact     = (G3 - compact_model)**2,
    res_augmented   = (G3 - augmented_model)**2
  )

(sse_c <- sum(errors$res_compact)) # 8190.777
(sse_a <- sum(errors$res_augmented)) # 7185.053
(ssr <- sse_c - sse_a) # 1005.724

(pre <- ssr / sse_c) # 0.1227874

(F <- (ssr / (2 - 1)) / (sse_a / (395 - 2))) # 55.00998

1 - pf(F, df1 = 1, df2 = 395 - 2) # 7.455148e-13
```

Source	SS	df	MS	F	p	PRE / R^2
failures	1005.724	1	1005.724	55.01	$< .001$	0.12
Error	7185.053	393	18.28258			
Total Error	8190.777	394				

Bericht aller Ergebnisse

Abschließend können wir alle Ergebnisse in einer Tabelle zusammen fassen:

Source	SS	df	MS	F	p	PRE / R^2
Regression	1084.856	2	542.43	29.59	$< .001$	0.13
studytime	10.6075	1	10.6075	0.58	0.447	0.00
failures	1005.724	1	1005.724	55.01	$< .001$	0.12
Error	7185.053	393	18.28258			
Total Error	8190.777	394				

Wir können also sagen, dass die Variable `failures` zu einer signifikanten Reduzierung des Fehlers führt und daher einen starken Beitrag macht, die Mathematikleistung der SuS zu erklären. Interessanterweise trägt die Variable `studytime` nicht zur Erklärung der Mathematikleistung bei. Wie viel Zeit SuS in das Lernen investieren, scheint daher keinen Einfluss auf deren Note zu haben. Man könnte sich im nächsten Schritt überlegen, welche anderen Variablen hilfreich wären, um die Mathematikleistung von SuS zu erklären. Was wir allerdings aus den Daten erkennen können, ist, dass das Vorwissen, welches in gewisser Weise durch die Variable `failures` abgedeckt ist, einen großen Einfluss auf zukünftiges Wissen hat. Dies ist ein Befund, den man immer wieder in der pädagogischen Psychologie findet.

Konfidenzintervalle

Erneut können wir die Signifkanz unserer Parameter durch Konfidenzintervalle testen. Die Berechnung ist genau gleich wie bei der einfachen Regression:

$$CI_{upper/lower} = b_i \pm \sqrt{\frac{F_{crit} * MSE}{(n-1) * s_x^2 * (1-R^2)}}$$

- MSE: Dies ist der Nenner der Formel des F-Tests: $F = \frac{SSR/(PA-PC)}{SSE(A)/(n-PA)} = \frac{MSR}{MSE}$
- s_x^2: Die Varianz der unabhängigen Variable (hier Lernzeit - Studytime).
- b_i: Der Steigungskoeffizient der unabhängigen Variable X_i.
- n: Die Anzahl der Untersuchungsobjekte.

- F_{crit}: Der kritische F-Wert, welcher zu einem signifikanten Ergebnis führt. Diesen kann in unserem Fall mit der Funktion `qf` berechnen: `qf(0.95, df1 = 1, df2 = 393)` $= 3.865229$.
- R^2: PRE, welches durch den Parameter aufgeklärt wird.

Zur Berechnung hilft uns unsere Tabelle:

Source	SS	df	MS	F	p	PRE / R^2
Regression	1084.856	2	542.43	29.59	$< .001$	0.13
studytime	10.6075	1	10.6075	0.58	0.447	0.00
failures	1005.724	1	1005.724	55.01	$< .001$	0.12
Error	7185.053	393	18.28258			
Total Error	8190.777	394				

Konfidenzintervall des Parameters Studytime

```
(ci_upper <- 0.1985 + sqrt((3.865229 * 18.28258) /
                           ((395 - 1) * var(student_data$studytime) * (1 - 0.00)))) # 0.703

(ci_lower <- 0.1985 - sqrt((3.865229 * 18.28258) /
                           ((395 - 1) * var(student_data$studytime)*  (1 - 0.00)))) # -0.30
```

Dies bedeutet, dass in 95 von 100 Fällen der wahre Steigungskoeffizient der Population sich in diesem Bereich befinden wird:

$$-0.36 \leq \beta_1 \leq 0.70$$

Da der Konfidenzintervall die 0 umschließt, wissen wir, dass es sich um ein nicht-signifikantes Ereignis handelt.

Konfienzintervall des Parameters Failures

Die gleiche Berechnung können wir für den Parameter failures aufstellen:

```
(ci_upper <- -2.1815 + sqrt((3.865229  * 18.28258) /
                           ((395 - 1) * var(student_data$failures) * (1 - 0.12)))) # -1.574

(ci_lower <- -2.1815 - sqrt((3.865229 * 18.28258) /
                           ((395 - 1) * var(student_data$failures) * (1 - 0.12)))) # -2.788
```

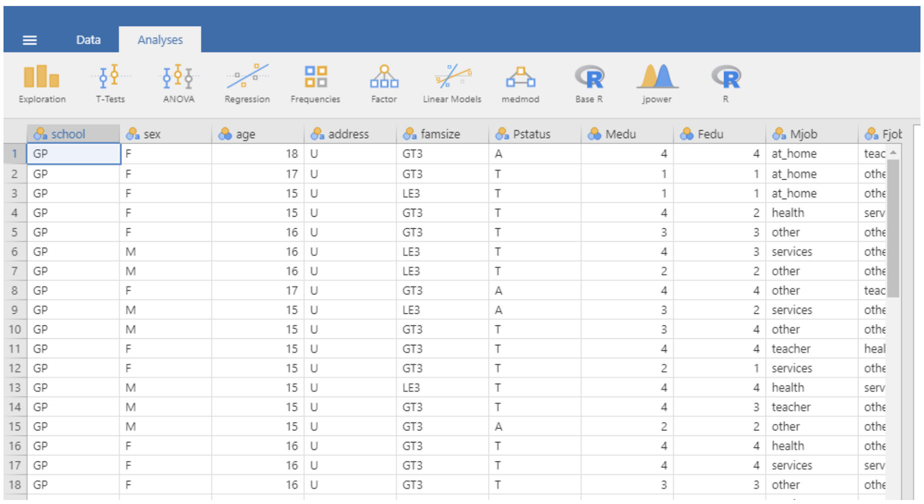
$$-2.79 \leq \beta_1 \leq -1.57$$

In 95 von 100 Fällen in denen wir demnach Konfidenzintervalle berechnen, wird sich der wahre Konfidenzintervall in diesem Bereich befinden. Wir sind demnach zuversichtlich, dass der Steigungskoeffizient der Variable failure demnach nicht 0 entspricht und daher dazu beiträgt, die Mathematikleistung der SuS zu erklären.

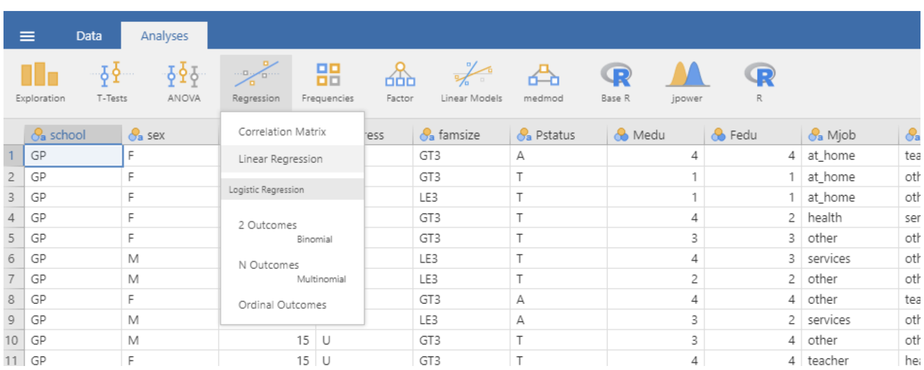
Computer-basierte Berechnung

Jamovi

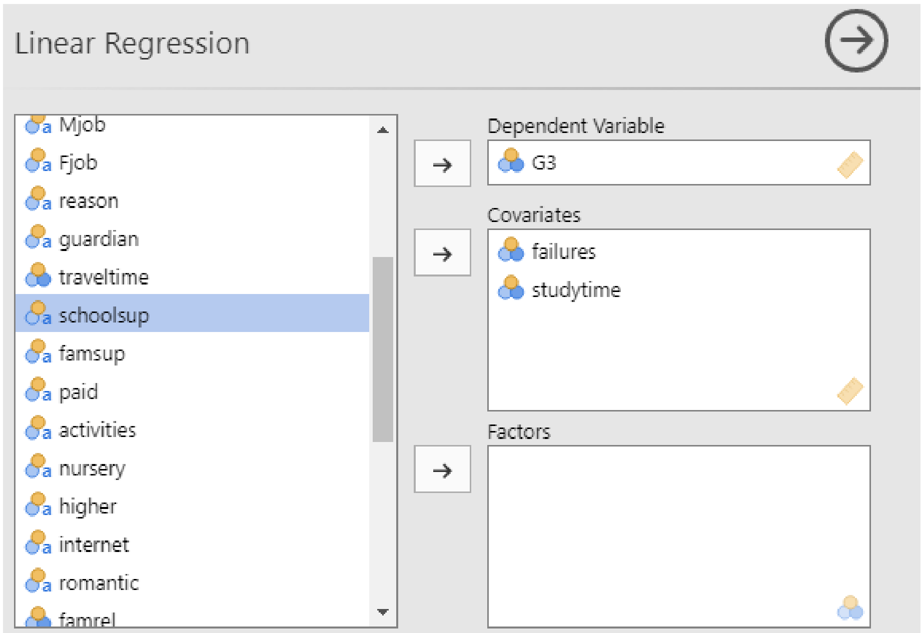
Um die multiple lineare Regression in Jamovi zu berechnen, müssen wir zunächst den Datensatz als CSV-Datei einlesen:



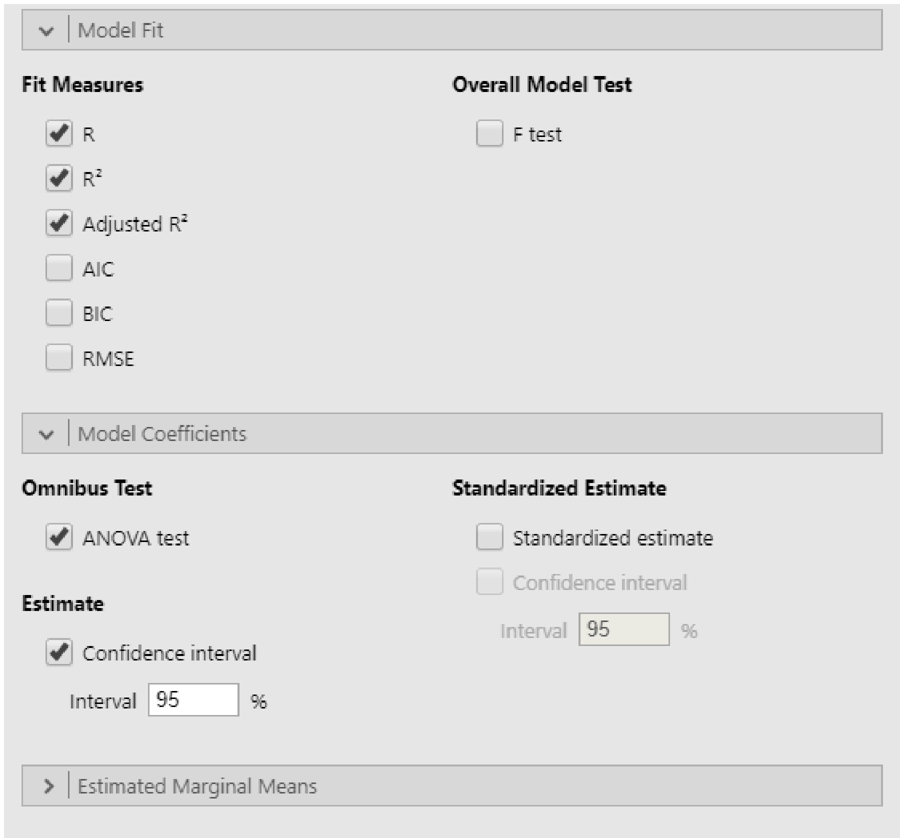
Anschließend kliegst du auf Regression -> Linear Regression:



Gebe nun die abhängige und unabhängie Variable an:



Lass dir zudem das adjustierte R^2, die Konfidenzintervalle und den ANOVA-Test ausgeben:



Anhand der Ergebnisse siehst du, dass wir die gleichen Ergebnisse behalten, die wir händisch berechnet haben:

Linear Regression

Model Fit Measures

Model	R	R^2	Adjusted R^2
1	0.362	0.131	0.127

Omnibus ANOVA Test

	Sum of Squares	df	Mean Square	F	p
failures	1005.7	1	1005.7	54.870	< .001
studytime	10.6	1	10.6	0.579	0.447
Residuals	7185.1	392	18.3		

Note. Type 3 sum of squares

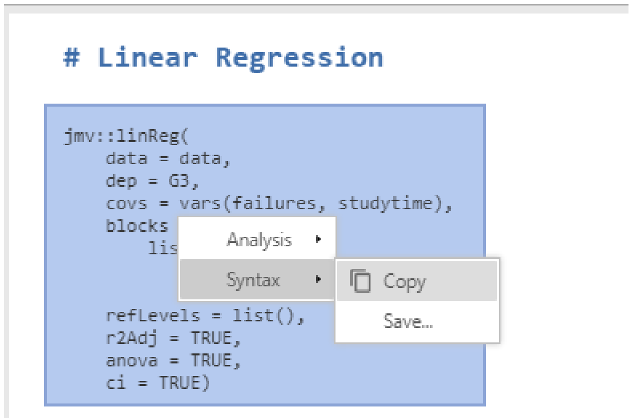
Model Coefficients

			95% Confidence Interval			
Predictor	Estimate	SE	Lower	Upper	t	p
Intercept	10.740	0.597	9.567	11.914	17.991	< .001
failures	−2.182	0.295	−2.761	−1.603	−7.407	< .001
studytime	0.198	0.261	−0.315	0.712	0.761	0.447

Source	SS	df	MS	F	p	PRE / R^2
Regression	1084.856	2	542.43	29.59	$< .001$	0.13
studytime	10.6075	1	10.6075	0.58	0.447	0.00
failures	1005.724	1	1005.724	55.01	$< .001$	0.12
Error	7185.053	393	18.28258			
Total Error	8190.777	394				

Zwar berechnet Jamovi einen t-Test für die Regressionskoeffizienten, du hast aber bereits gesehen, wie du die F-Werte in T-Werte umwandeln kannst. Für die Interpretation der Ergebnisse ist dieser Unterschied unerheblich.

Im nächsten Schritt kopierst du den R-Code:



Diesen fügst du nun in R ein und änderst die Bezeichnung des Datensatzes:

```r
library(jmv)

# Verfahren: Multiple lineare Regression
# AV: G3 - Mathematikleistung (intervalskaliert)
# UV:
#    - studytime (intervallskaliert)
#    - freedom   (intervallskaliert)
jmv::linReg(
  data = student_data,
  dep = G3,
  covs = vars(failures, studytime),
  blocks = list(
    list(
      "failures",
      "studytime")),
  refLevels = list(),
  r2Adj = TRUE,
  anova = TRUE,
  ci = TRUE)
```

17:16 (Top Level) ÷

Console Terminal ×

C:/Users/ChristianEZW/Downloads/

```
> library(jmv)
>
> jmv::linReg(
+   data = student_data,
+   dep = G3,
+   covs = vars(failures, studytime),
+   blocks = list(
+     list(
+       "failures",
+       "studytime")),
+   refLevels = list(),
+   r2Adj = TRUE,
+   anova = TRUE,
+   ci = TRUE)
```

LINEAR REGRESSION

Model Fit Measures

Model	R	R²	Adjusted R²
1	0.362	0.131	0.127

MODEL SPECIFIC RESULTS

MODEL 1

Omnibus ANOVA Test

	Sum of Squares	df	Mean Square	F	p
failures	1005.7	1	1005.7	54.870	< .001
studytime	10.6	1	10.6	0.579	0.447
Residuals	7185.1	392	18.3		

Note. Type 3 sum of squares

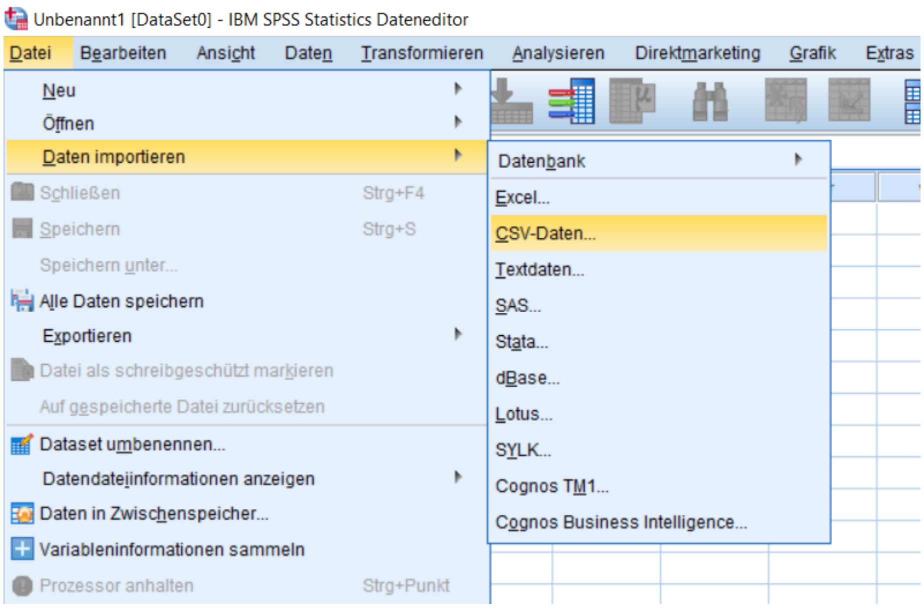
Model Coefficients

Predictor	Estimate	SE	Lower	Upper	t	p
Intercept	10.740	0.597	9.567	11.914	17.991	< .001
failures	-2.182	0.295	-2.761	-1.603	-7.407	< .001
studytime	0.198	0.261	-0.315	0.712	0.761	0.447

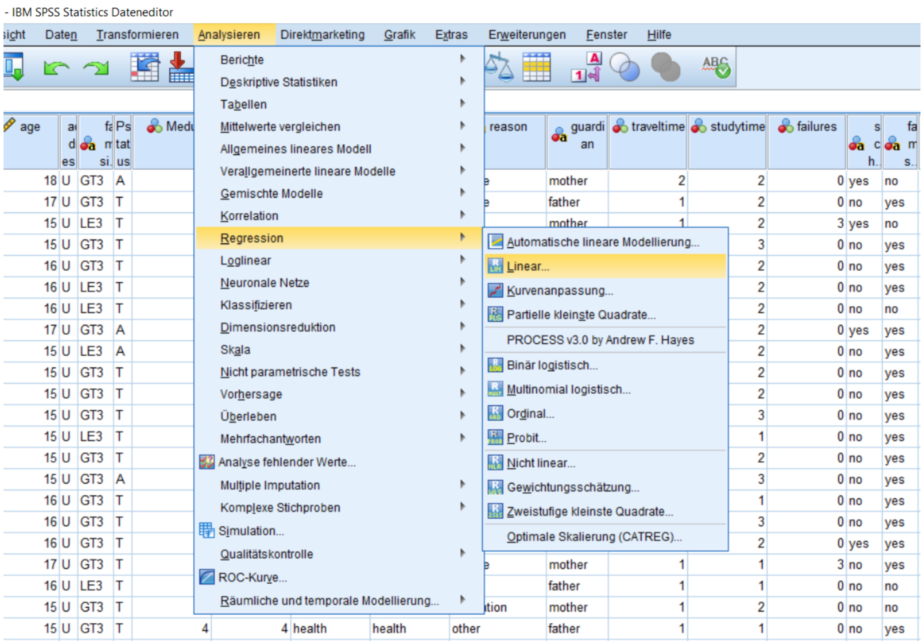
Achte darauf, dass du das Verfahren durch Kommentare dokumentierst, so dass du später weißt, was du gerechnet hast.

SPSS

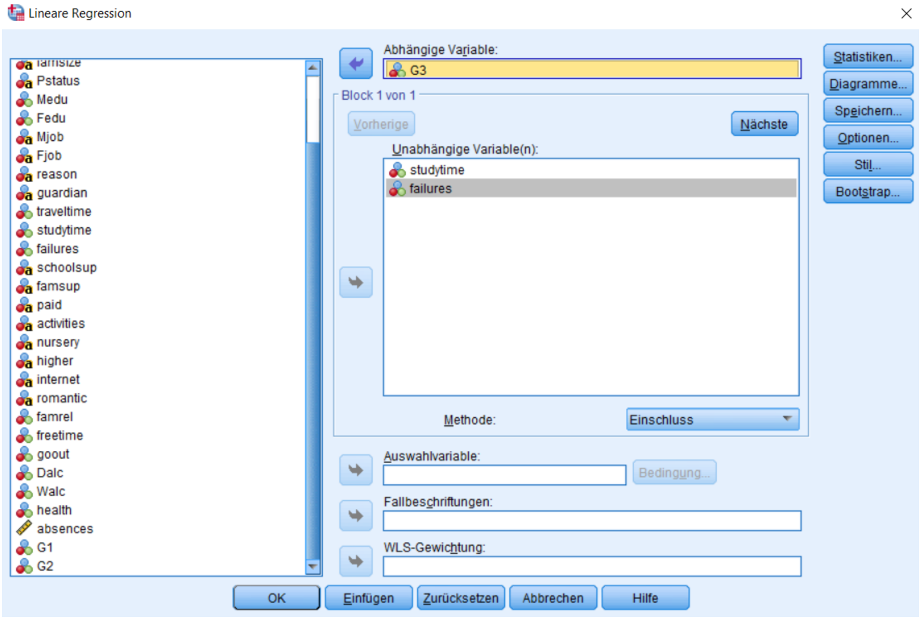
In SPSS importierst du zunächst die CSV-Datei:



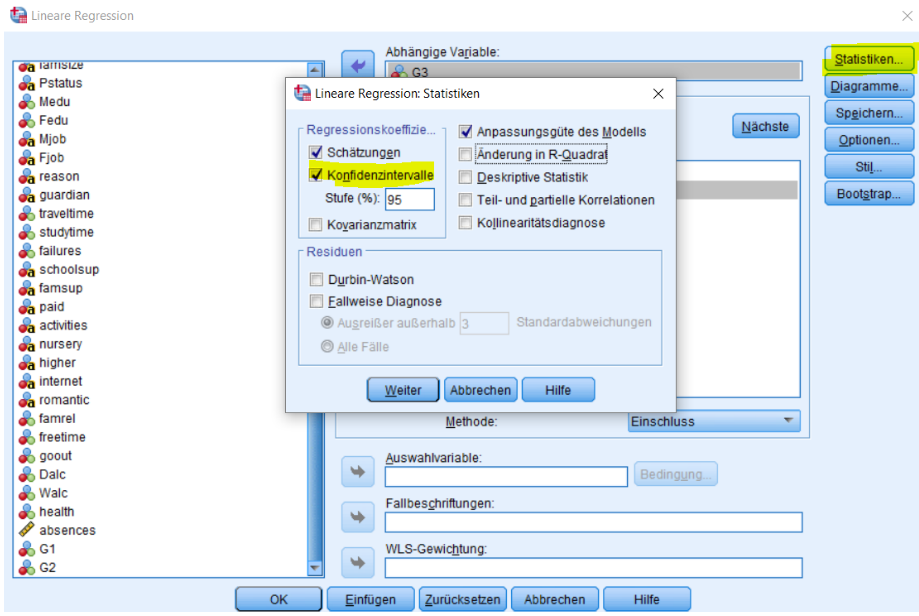
Anschließend wählst du Regression -> Linear aus:



Im Anschluss bestimsmt du die abhängige und die unabhängigen Variablen:



Unter Statistiken füge die Konfidenzintervalle hinzu:



Drucke ok. Du erhältst nun folgenden Output:

Modellzusammenfassung

Modell	R	R-Quadrat	Korrigiertes R-Quadrat	Standardfehler des Schätzers
1	.362[a]	.131	.127	4.281

a. Einflußvariablen : (Konstante), failures, studytime

ANOVA[a]

Modell		Quadratsumme	df	Mittel der Quadrate	F	Sig.
1	Regression	1084.856	2	542.428	29.594	.000[b]
	Nicht standardisierte Residuen	7185.053	392	18.329		
	Gesamt	8269.909	394			

a. Abhängige Variable: G3

b. Einflußvariablen : (Konstante), failures, studytime

Koeffizienten[a]

Modell		Nicht standardisierte Koeffizienten		Standardisierte Koeffizienten			95,0% Konfidenzintervalle für B	
		Regressions koeffizientB	Standardfehler	Beta	T	Sig.	Untergrenze	Obergrenze
1	(Konstante)	10.740	.597		17.991	.000	9.567	11.914
	studytime	.198	.261	.036	.761	.447	-.315	.712
	failures	-2.182	.295	-.354	-7.407	.000	-2.761	-1.603

a. Abhängige Variable: G3

Verglichen mit dem Output von Jamovi sind die Daten identisch:

Linear Regression

Model Fit Measures

Model	R	R^2	Adjusted R^2
1	0.362	0.131	0.127

Omnibus ANOVA Test

	Sum of Squares	df	Mean Square	F	p
failures	1005.7	1	1005.7	54.870	< .001
studytime	10.6	1	10.6	0.579	0.447
Residuals	7185.1	392	18.3		

Note. Type 3 sum of squares

Model Coefficients

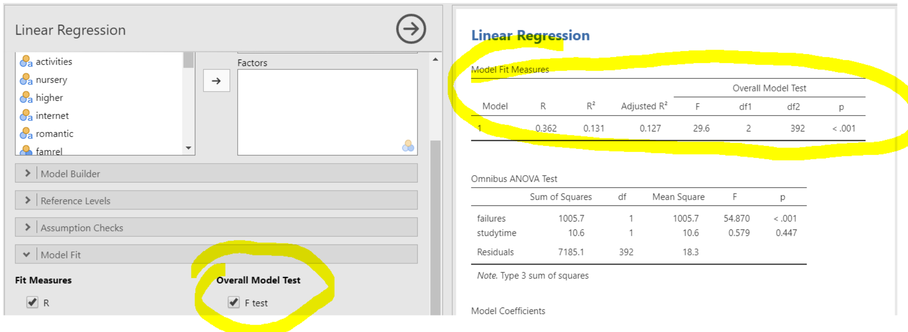
			95% Confidence Interval			
Predictor	Estimate	SE	Lower	Upper	t	p
Intercept	10.740	0.597	9.567	11.914	17.991	< .001
failures	−2.182	0.295	−2.761	−1.603	−7.407	< .001
studytime	0.198	0.261	−0.315	0.712	0.761	0.447

Lediglich der ANOVA-Test unterscheidet sich (zweite Tabelle in SPSS). Dieser beschreibt das erste Modell, welches wir in diesem Modul gerechnet hatten:

$$MODEL\ A = Y_i = \beta_0 + \beta_1 * X_1 + \beta_2 * X_2 + \epsilon_i$$
$$MODEL\ C = Y_i = B_0 + \epsilon_i$$

Source	SS	df	MS	F	p	PRE / R^2
Reduction	1084.856	2	542.43	29.59	$< .001$	0.13
Error	7185.053	392	18.33			
Total Error	8269.909	394				

In Jamovi können wir uns diesen auch ausgeben lassen: Model Fit -> Overall Model Test -> F test:



R

In R können wir die gleiche Berechnung durch die Funktion `lm` durchführen:

```
lm(G3 ~ studytime + failures, data = student_data) %>%
  summary

```

```
3:1    (Top Level) ÷
Console   Terminal ×
C:/Users/ChristianEZW/Downloads/
> lm(G3 ~ studytime + failures, data = student_data) %>%
+   summary

Call:
lm(formula = G3 ~ studytime + failures, data = student_data)

Residuals:
     Min       1Q   Median       3Q      Max 
-11.5342  -1.9556   0.0613   3.0359   9.2429 

Coefficients:
            Estimate Std. Error t value Pr(>|t|)    
(Intercept)  10.7402     0.5970  17.991  < 2e-16 ***
studytime     0.1985     0.2610   0.761    0.447    
failures     -2.1815     0.2945  -7.407 7.97e-13 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 4.281 on 392 degrees of freedom
Multiple R-squared:  0.1312,	Adjusted R-squared:  0.1267 
F-statistic: 29.59 on 2 and 392 DF,  p-value: 1.072e-12
```

Multiple Regression berichten

Wenn wir unser Ergebnis nun ein einem Artikel berichten möchten, können wir dies folgendermaßen tun:

> Es wurde eine multiple Regression mit der Mathematikleistung als abhängige Variable und der Durchfallquote sowie der Lernzeit als unabhängige Variable berechnet. Die Regression ergab einen signifikanten Effekt der beiden Prädiktoren, $F(2, 392) = 29.59$, $p < .001$, R^2 = .13. Die Untersuchung der einzelnen Prädiktoren ergab, dass der Prädiktor Durchfallquote einen signifikanten Effekt auf die Mathematikleistung der Schüler*innen hat, $F(1, 392) = 54.87$, $p < .001$, was darauf hindeutet, dass die Durchfallquote die Mathematikleistung der Schüler*innen negativ beeinflusst. Für den Prädiktor Lernzeit ergab sich kein signifikanter Effekt, $F(1, 392) = 0.58$, $p = .48$, was darauf hindeutet, dass die Mathematikleistung nicht von der Lernzeit beeeinflusst wird.

Interpretation der multiplen Regression

Kausale Aussagen

Wir hatten die partiellen Regressionskoeffizienten als die Veränderung in der abhängigen Variable beschrieben, die auftreten, wenn wir für alle anderen Prädiktoren kontrollieren. Diese Tatsache bedeutet allerdings **nicht**, dass die abhängige Variable durch die unabhängige Variable verändert wird. Die multiple Regression beschreibt lediglich die Daten. Beispielsweise können wir auf Grundlage der multiplen Regression nicht behaupten, dass eine höhere Durchfallquote zu einer schlechteren Mathematikleistung führt; selbst wenn der Regressionskoeffizient negativ ist.

Hier findest du eine Webseite, die mehrere irrsinnige kausale Aussagen zweier Variablen veranschaulicht. Beispielsweise gibt es einen nachweisbaren negativen Zusammenhang zwischen der Verkauf von Eis in einer Stadt und den Selbstmorden in einer Stadt. Führt weniger Eiskaufen kaufen daher zu Selbstmord? Nein. Der Grund liegt vielmehr in einer dritten Variable, der Temperatur. Die Temperatur wiederum könnte auf die Stimmung von Personen wirken, da es im Winter weniger Licht gibt.

Wir werden im nächsten Modul allerdings ein Design / ein Modell kennen lernen, auf Grund dessen wir kausale Aussagen treffen können. Diese Designs sind fast immer Experimente, bei denen wir eine Variable bewusst manipulieren, um ihren Effekt zu bestimmen.

Wichtigkeit der Prädiktoren

Ein häufiger Fehler in der Interpretation einer multiplen Regression liegt darin, dass die Stärke der Prädiktoren falsch interpretiert wird. Schauen wir uns dazu erneut unser Modell an:

$$\hat{Y}_i = 10.74 - 2.18 * X_{i1} + 0.19 * X_{i2} + e_i$$

Der Regressionskoeffizient des Durchfallens b_1 liegt bei -2.18. Der Regressionskoeffizient der Lernzeit b_2 liegt bei 0.19. Mehr Durchfallen führt daher zu einer schlechteren Mathemleistung, mehr Lernen zu einer leicht besseren, allerdings ist dieser Prädiktor nicht signifikant.

Es wäre nun inkorrekt zu behaupten, dass die Durchfallquote einen stärkeren Einfluss auf die Mathematikleistung hat als die Lernzeit. Hättest du aus irgendwelchen Gründen beispielsweise die Variable `failures` durch 1000 geteilt, wäre der Regressionskoeffizient b_1 tausendfach kleiner. Der Beitrag auf die abhängige Variable hingegen bliebe gleich.

Häufig werden die Variablen daher z-standardisiert, um ihre Interpretation zu ermöglichen. Auch dieses Vorgehen ist nicht empfehlenswert, da diese von der Streuung der Variable abhängig sind. Zudem löst die Standardisierung nicht das Problem der Redundaz. Die Stärke der Regressionskoeffizienten sind daher mit Vorsicht zu genießen und sollten nicht überinterpretiert werden.

Modeling

Modeling