Logistische Regression

Richard Hunger

08. August 2024

Übersicht

- 4-Felder Tafeln und Odds Ratio
- Logistisches Regressionsmodell
- Logistische Funktion
- Prädiktion
- Koeffizientenprüfung
- Residuen
- Ausreißerdiagnostik
- kategoriale Prädiktoren
- Multiple Logistische Regression
- Modellvergleiche
- Interaktionen
- Modellgüte
- Ergebnisbericht
- Übungen

Einordnung

- **Prädiktion** Basierend auf bisherigen Beobachtungsdaten soll eine Vorhersage für bisher unbekannte Beobachtungseinheiten getroffen werden
- 2 Gruppen von Prädiktionsmodellen:
- *Regressionsmodelle*: konkreten numerischen Wert vorhersagen
  - Zielvariable: numerisch
  - z.B. Hauspreis, Lebenserwartung, Verkaufszahlen
  - Modelliert wird ein konkreter Wert
- *Klassifikationsmodelle*: Gruppenzugehörigkeit vorhersagen
  - Zielvariable: kategorial
  - z.B. Ansprechen auf Therapie, Kreditausfall, Spam-Mail
  - Modelliert wird eine Wahrscheinlichkeit
- Klassifikation: **Logistische Regression**

Vgl. mit OLS

- **Lineare Regression**
- metrisches Kriterium
- modelliert wird der Kriteriumswert

- **Logistische Regression**
- binäres/dichotomes Kriterium
- modelliert wird die Wahrscheinlichkeit

- lineare Regression nicht möglich
  - keine NV-Residuen
  - keine Varianzhomogenität
  - Prognosen außerhalb plausiblen Bereichs
- Gemeinsamkeiten OLS:
  - Globaltest (ANOVA) zur Modellsignifikanz
  - Einzeltests zu den Parametern
  - Bestimmtheitsmaße
  - Prädiktorenvielfalt

Logistische Regression

- **Logistische Regression**
  - AV: dichotome/binäre Variable (0/1-Kodierung)
  - UVs: metrische und/oder kategoriale Merkmale
- Beispiele:
  - Kaufentscheidung für ein Produkt
  - Auftreten einer Krankheit
  - Kreditausfall
  - Erfolg eines Produktes
- für die LR werden ORs benötigt…

4-Felder Tafeln

Risiko

- Grundlage 4-Felder Tafel (Kontingenztabelle)
- 2 dichotome Variablen
- UV und AV
- Beispiel: Studie zu Aspirineinnahme und Myokardinfarkt
-

	Myokardinfarkt		
	Ja	Nein	TOTAL
Gruppe			
Placebo	189	10'845	11'034
Aspirin	104	10'933	11'037
TOTAL	293	21'778	22'071

- $\hat{\pi}_1 = 189/11034 = 0.0171 = 1.7\%$
- $\hat{\pi}_2 = 104/11037 = 0.0094 = 0.9\%$

4-Felder Tafeln

Risikodifferenz

- $\hat{\pi}_1 = 189/11034 = 0.0171 = 1.7\%$
- $\hat{\pi}_2 = 104/11037 = 0.0094 = 0.9\%$
- absolute Risikodifferenz:
  - $\hat{\pi}_1 - \hat{\pi}_2 = 0.0077 = 0.8\%$
- mit 95%-KI:

```
DescTools::BinomDiffCI(189,11034,104,11037, conf.level = .95) %>% flex()
```

	Var2		
	est	lwr.ci	upr.ci
Var1 A	0.007706024	0.004676768	0.01073254

- KI enthält nicht die 0, also schützt Aspirin vor MI!

4-Felder Tafeln

relatives Risiko

- Differenzen zw. Anteilen aussagekräftiger bei extremen Ausprägungen (nahe 0 bzw. 1)
- Beispiel:
  - 0.010 zu 0.001 vs. 0.410 zu 0.401
  - in beiden Fällen Differenz von 0.009
- ggf. aussagekräftiger: **relatives Risiko**
  - *risk ratio*
  - $RR = \pi_1 / \pi_2$
  - RR = 0.010 / 0.001 = 10
  - RR = 0.410 / 0.401 = 1.02
- MI-Studie:
  - 0.0171 / 0.0094 = 1.82
  - 82% erhöhtes MI Risiko in Placebo Gruppe
  - vs. 0.77% absolute Differenz

4-Felder Tafeln

relatives Risiko

- in R mit 95%-KI berechenbar:

```
df_chisq[1:2, 1:2] %>% DescTools::RelRisk(conf.level = .95) %>% enframe() %>% flex()
```

name	value
rel. risk	1.82
lwr.ci	1.43
upr.ci	2.30

- KI enthält nicht die 1, also schützt Aspirin vor MI!

4-Felder Tafeln

Odds

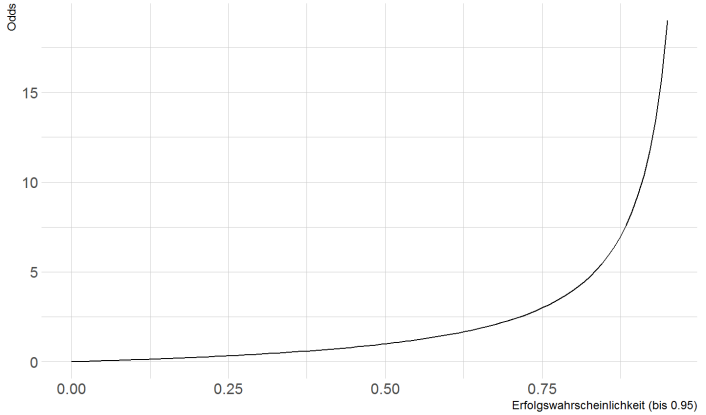
- weiteres Maß: **Odds**
- = Chancenverhältnis, Gewinnquote
- gebräuchlich bei Pferderennen, Sportwetten
- Berechnung:
  - Erfolgswahrscheinlichkeit / (1-Erfolgswahrscheinlichkeit)
  - $Odds = \pi/(1-\pi)$
  - “Erfolg” = interessierender Outcome
- zB
  - Würfeln: $P(x=6) = \frac{1}{6} = 0.166$
  - $Odds = \frac{1/6}{5/6} = 0.20$
  - $Odds = \frac{1}{5} = 0.20$
- Interpretation:
  - Verhältnis $\frac{1}{5}$: 1 “Erfolg” bei 5 “Fehlern”

4-Felder Tafeln

Odds

- weiteres Beispiel:
  - $\pi = 0.75$
  - Odds?
  - $Odds = \frac{0.75}{0.25} = \frac{3}{1} = 3.0$
  - 3 “Erfolge” pro 1 “Fehler”
- Allg. Interpretation:
  - Odds = 1 … gleich häufige Ereignisse (Baseline)
  - Odds < 1 … Erfolg seltener
  - Odds > 1 … Erfolg häufiger
- Wertebereich: $[0; +\infty]$

- Zusammenhang:



4-Felder Tafeln

Odds

- Umrechnung von Odds in P
  - $\pi = \frac{Odds}{Odds+1}$
- Bsp:
  - $\pi = 0.80$
  - $Odds = \frac{0.8}{0.2} = 4.0$
  - $\pi = \frac{4.0}{4.0+1.0} = \frac{4}{5} = 0.80$
- MI-Beispiel:

	Myokardinfarkt		
	Ja	Nein	TOTAL
Gruppe			
Placebo	189	10'845	11'034
Aspirin	104	10'933	11'037
TOTAL	293	21'778	22'071

  - $Odds_{Placebo} = 189/10845 = 0.0174$
  - $Odds_{Aspirin} = 104/10933 = 0.0095$

4-Felder Tafeln

Odds Ratio

- Analog zum relativen Risiko: *Odds Ratio*
  - $\theta = \frac{Odds_1}{Odds_2} = \frac{\pi_1/(1-\pi_1)}{\pi_2/(1-\pi_2)}$
- MI-Beispiel:

	Myokardinfarkt		
	Ja	Nein	TOTAL
Gruppe			
Placebo	189	10'845	11'034
Aspirin	104	10'933	11'037
TOTAL	293	21'778	22'071

  - $OR = \frac{0.0174}{0.0095} = 1.83$
  - Das Risiko für MI ist in der Placebo-Gruppe 1.83-fach erhöht bzw. 83% höher.
- in R:

```
df_chisq[1:2,1:2] %>% DescTools::OddsRatio(conf.level = .95) %>% enframe() %>% flex()
```

name	value
odds ratio	1.83
lwr.ci	1.44
upr.ci	2.33

4-Felder Tafeln

Odds Ratio

- Übersicht:

Exposure Status	Event Occurred	
	Yes	No
Exposed	a	b
Not Exposed	c	d

$$\text{Relative Risk} = \frac{a/(a+b)}{c/(c+d)}$$

$$\text{Odds Ratio} = \frac{a/b}{c/d} = \frac{ad}{cb}$$

- **Odds Ratio** = Zusammenhangsmaß für 4-Felder Tafeln
- Reihenfolge der Kategorien (Zeilen bzw. Spalten) beliebig
- Inverse Beziehung:
  - OR=4 identisch zu OR=0.25
  - OR=5 identisch zu OR=0.20
- Odds = 1 … Referenzwert für statistische Testung
- Wertebereich: $[0; +\infty]$

4-Felder Tafeln

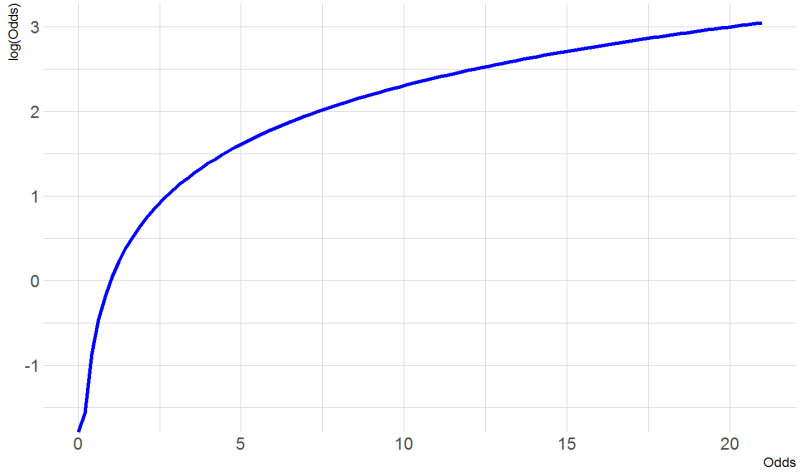
Log Odds

- Odds sind sehr schief verteilt:
  - Bereich Odds<1 nur sehr schmal
  - Bereich Odds>1 sehr breit
- durch Logarithmierung:
  - aus WB: $[0; +\infty]$ wird
  - $[-\infty; +\infty]$
- Berechnung logarithmierte Oddss
  - $\ln(Odds) = \log_{2.718}(Odds) = \log_e(Odds)$
  - üblich ist Basis e[1]
- Eigenschaften:
  - näher an Normalverteilung
  - symmetrische Verteilung!
  - Baseline Odds = 1 wird zu log(Odds) = 0

4-Felder Tafeln

Log Odds

- Bsp:
  - Odds = 5.0, log(5.0) = 1.61
  - Odds = 0.2, log(0.2) = -1.61
- Zusammenhang Odds und log(Odds), graphisch:



- Inferenz basiert auf logarithmierten ORs
  - Berechnung bspw. von KI auf logarithmierten Werten und Rücktransformation
  - Rücktransformation: $Odds = e^{log(Odds)}$

4-Felder Tafeln

Log Odds

- Zusammenhang zwischen OR und RR
  - $OR = \frac{\pi_1/(1-\pi_1)}{\pi_2/(1-\pi_2)} = RR \times \left(\frac{1-\pi_2}{1-\pi_1}\right)$
- bei kleinen Werten von π_1 und π_2 gilt $OR \approx RR$
  - MI-Beispiel: RR = 1.82 und OR = 1.83

4-Felder Tafeln

Aufgabe 1

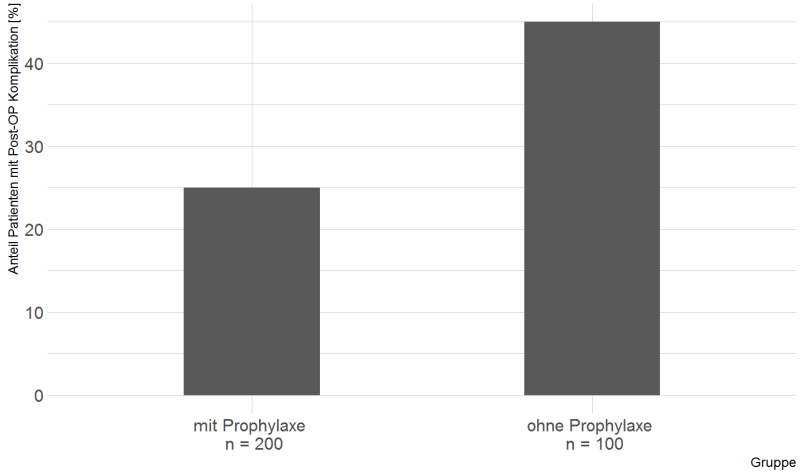
- In einer Studie wurden Patienten mit einer koronaren Gefäßerkrankung, denen ein Stent implantiert wurde, eingeschlossen. Von 104 Patienten, die zur vereinbarten Kontrolluntersuchung nach 6 Monaten erschienen, hatten 40 keine Symptome einer Angina pectoris. Weitere 49 Patienten, die nicht zum Kontrolltermin erschienen, wurden zu Hause besucht. Dabei wurde festgestellt, dass 33 von ihnen asymptomatisch waren. Stellen Sie die gegebenen Werte in einer Kontingenztafel dar und berechnen Sie die das Risiko, die Risikodifferent, das relative Risiko, die Odds und das Odds Ratio, das den Zusammenhang zwischen Erscheinen zur Kontrolluntersuchung und dem klinischen Zustand des Patienten beschreibt. Beachten Sie bei der Interpretation die Richtung der Zusammenhänge.

```
df_or <- matrix(c(40,64, 33,16), byrow = T,
       dimnames = list(Ort = c('Klinik', 'Heim'), Symptom = c('Ja', 'Nein')), 2, 2)

DescTools::BinomDiffCI(40, 104, 33, 49, conf.level = .95)
df_or %>% DescTools::RelRisk(conf.level = .95)
df_or %>% DescTools::OddsRatio(conf.level = .95)
```

4-Felder Tafeln

Aufgabe 2

- Zur Vermeidung von postoperativen Komplikationen wurde eine Studie durchgeführt. Dafür wurden zwei Gruppen vergleichen. Eine davon erhielt intaoperativ 10 mg Dexamethason während die andere die Standardversorgung erhielt. Die Ergebnisse sind in folgender Abbildung zusammengefasst:
-

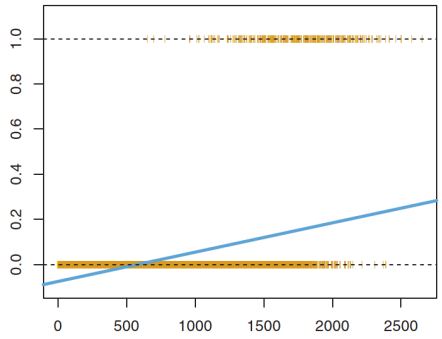


- Berechnen Sie wieder Risiko, die Risikodifferenz, das Risikoverhältnis, die Odds und das Odds Ratio.

Logistisches Regressionsmodell

Grundlagen

- Prinzipiell geht es um die Modellierung des Eintreten eines Ereignisses $P(Y = 1) = \pi$ (Erfolg)
- oder ebend nicht $P(Y = 0) = 1 - \pi$ (Fehler)
- dabei gilt $E(Y) = \pi$
- lineares Regressionsmodell funktioniert nicht:

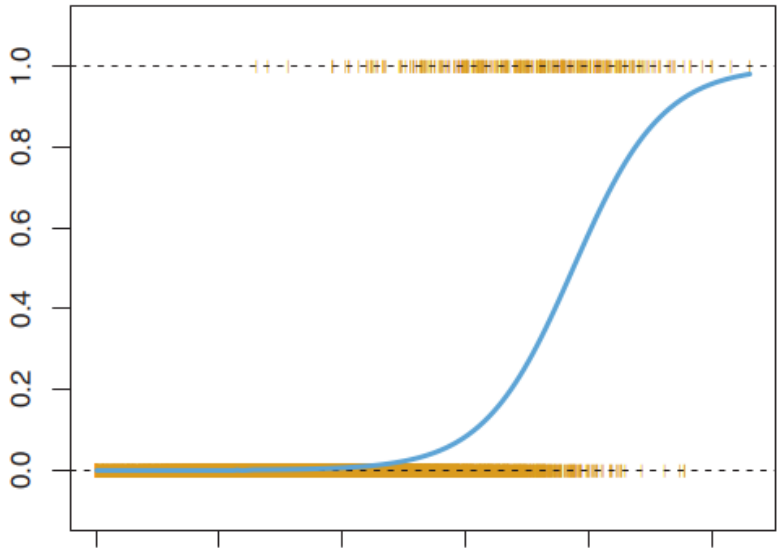


- ⚠ Probleme
  - negative Wahrscheinlichkeitswerte
  - bei hohen Prädiktorwerten zu niedrige Wk geschätzt
  - nicht-lineare Effekte der UVs auf AV (Bsp. Einkommen und Neuwagenkauf)

Logistisches Regressionsmodell

Grundlagen

- Die Ergebnisse des linearen Wahrscheinlichkeitsmodells sollen irgendwie in den Bereich von 0 bis 1 begrenzt werden.
- Dabei idR streng monotoner Zusammenhang zwischen X und P(Y=1): je mehr X um so (un-) wahrscheinlicher Y=1
- Also brauchen wir so was:



Logistisches Regressionsmodell

Grundlagen

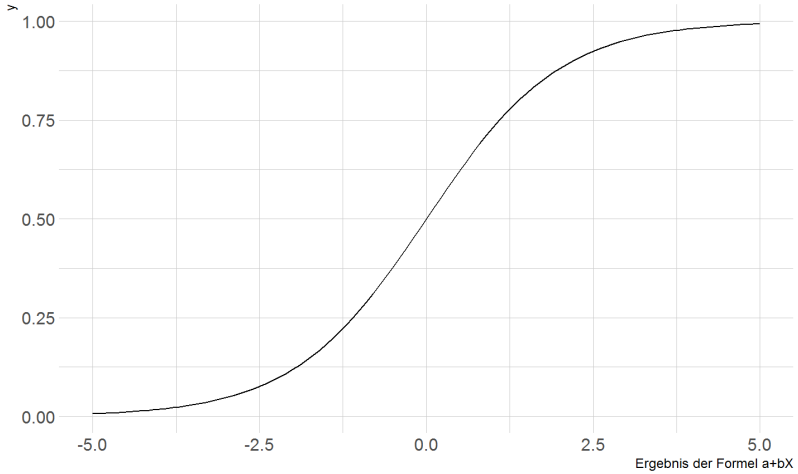
- eine bestimmte Funktion erfüllt diese Anforderung
  - **logistische Funktion**
  - $P(Y = 1) = \pi = \frac{\exp(\alpha+\beta x)}{1+\exp(\alpha+\beta x)}$
- Vgl. Umrechnung von Odds in Wahrscheinlichkeit:
  - $\pi = \frac{Odds}{1+Odds}$
  - Bereich von 0 bis ∞ wird auf [0,1] normiert
- Beispiele:
  - $exp(-5)/(exp(-5)+1) = 0.007$
  - $exp(-1)/(exp(-1)+1) = 0.268$
  - $exp(0)/(exp(0)+1) = 0.500$
  - $exp(1)/(exp(1)+1) = 0.731$
  - $exp(5)/(exp(5)+1) = 0.993$

Logistisches Regressionsmodell

Grundlagen

- logistische Funktion (graphisch):

```
ggplot() +
  geom_function(fun = ~exp(.x)/(1+exp(.x)), xlim = c(-5,5)) +
  labs(x = "Ergebnis der Formel a+bX")
```

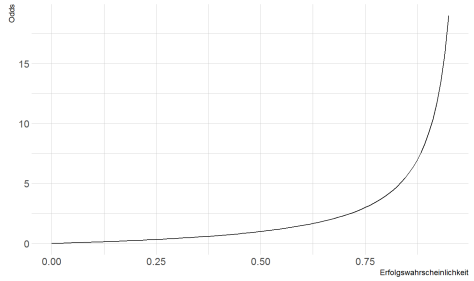


- beliebig große (kleine) x-Werte resultieren immer in Werten im Bereich von $]0;1[$
- ↯ Aber noch keine lineare Beziehung zw. x und y

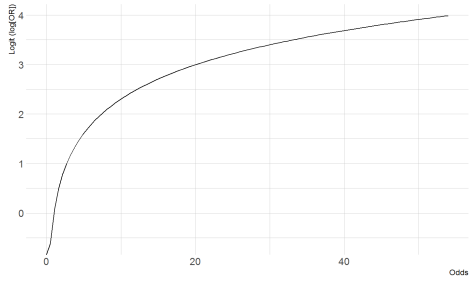
Logistisches Regressionsmodell

Grundlagen

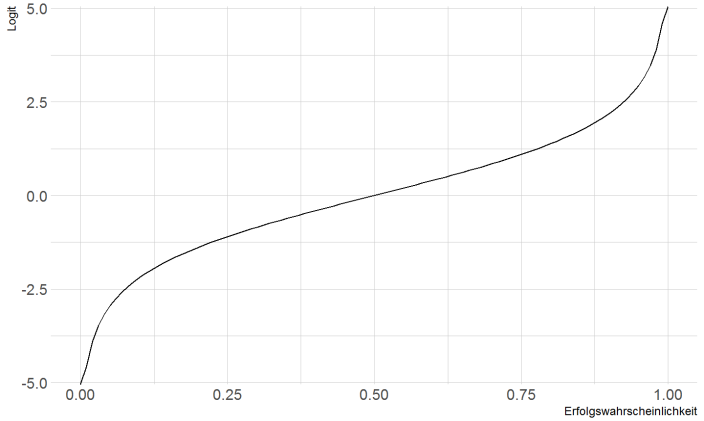
- Umstellen der logistischen Funktion: $\frac{\pi}{1-\pi} = e^{\alpha+\beta X}$
- Der Teil: $\frac{\pi}{1-\pi}$ waren die Odds!
- Recap: Odds $]0; +\infty]$



- Odds und Logits $[-\infty; +\infty]$



- Erfolgswahrscheinlichkeit und Logits:



Logistisches Regressionsmodell

Grundlagen

- umgestellte Funktion: $\frac{\pi}{1-\pi} = e^{\alpha+\beta X}$
- nun beide Seiten logarithmieren, dann erhält man:
- $\log\left[\frac{\pi(x)}{1-\pi(x)}\right] = \alpha + \beta X$
- Der Teil: $\log\left[\frac{\pi(x)}{1-\pi(x)}\right]$ sind also logarithmierte Odds
- auch log-Odds oder **Logits**
- Während Odds im Bereich $[0; +\infty]$ liegen
- können Logits Werte im Bereich $[-\infty; +\infty]$ annehmen

Umrechnungen

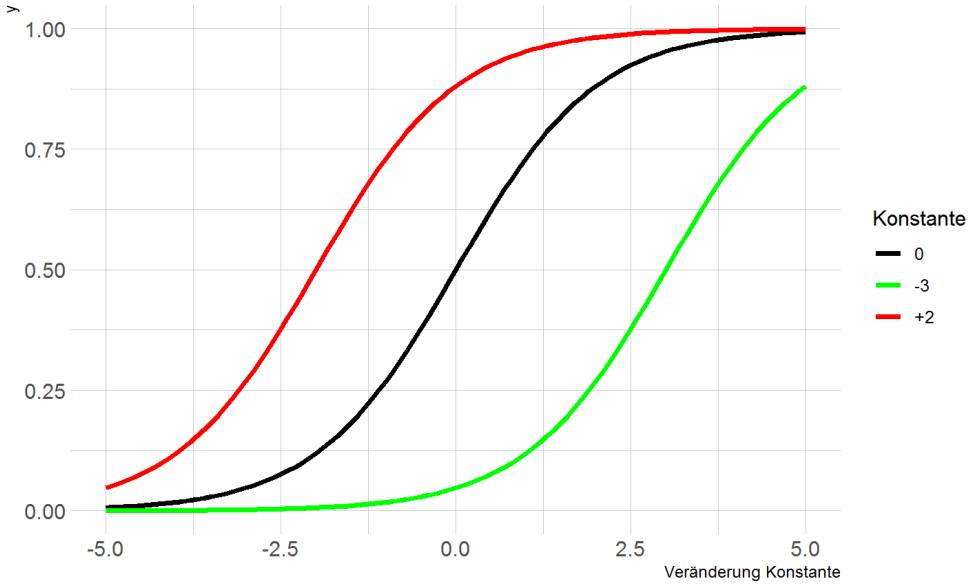
P, Odds, Logit

- nochmal von Hand 1:
  - $\pi = 0.8$
  - $Odds = 0.8/0.2 = 4.00$
  - $Logit = log(4.00) = 1.39$
- nochmal von Hand 2:
  - $\pi = 0.5$
  - $Odds = 0.5/0.5 = 1.00$
  - $Logit = log(1.00) = 0.00$
- nochmal von Hand 3:
  - $\pi = 0.3$
  - $Odds = 0.3/0.7 = 0.43$
  - $Logit = log(0.43) = -0.84$
- nochmal von Hand 4: (vgl. mit Bsp 1)
  - $\pi = 0.2$
  - $Odds = 0.2/0.8 = 0.25$
  - $Logit = log(0.43) = -1.39$

Funktionsverlauf

Intercept

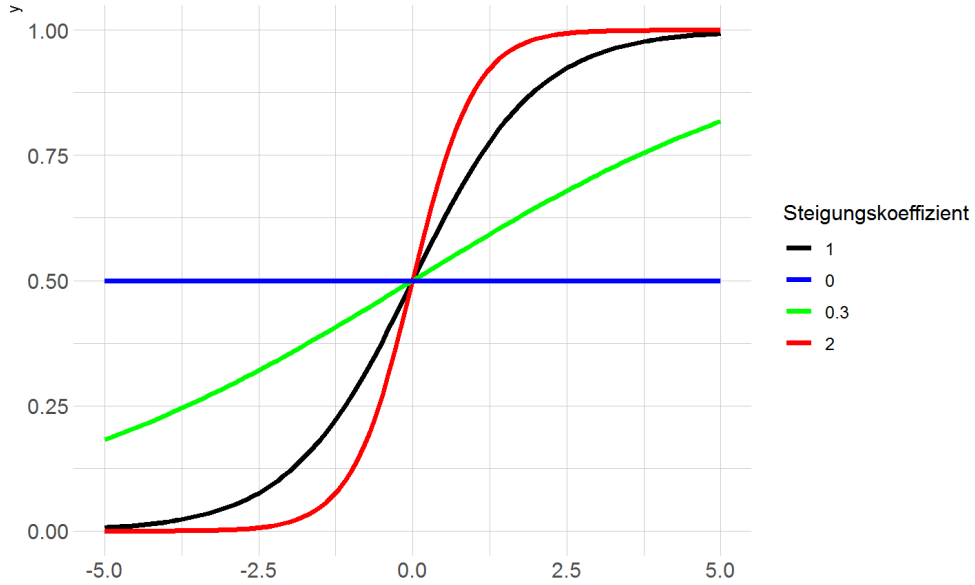
```
ggplot() +
  geom_function(fun = ~exp(.x)/(1+exp(.x)), xlim = c(-5,5),      aes(color = 'black'), linewidth = 1.3) +
  geom_function(fun = ~exp(.x+2)/(1+exp(.x+2)), xlim = c(-5,5), aes(color = 'red'  ), linewidth = 1.3) +
  geom_function(fun = ~exp(.x-3)/(1+exp(.x-3)), xlim = c(-5,5), aes(color = 'green'), linewidth = 1.3) +
  labs(x = "Veränderung Konstante") +
  scale_color_manual(name="Konstante",
                     labels=c("0","-3","+2"),
                     values=c("black","green","red"))
```



Funktionsverlauf

Steigung

```
ggplot() +
  geom_function(fun = ~exp(.x)/(1+exp(.x)), xlim = c(-5,5),      aes(color = 'black'), linewidth = 1.3) +
  geom_function(fun = ~exp(.x*2)/(1+exp(.x*2)), xlim = c(-5,5), aes(color = 'red'   ), linewidth = 1.3) +
  geom_function(fun = ~exp(.x*0.3)/(1+exp(.x*0.3)), xlim = c(-5,5), aes(color = 'green'), linewidth = 1.3) +
  geom_function(fun = ~exp(.x*0)/(1+exp(.x*0)), xlim = c(-5,5), aes(color = 'blue'), linewidth = 1.3) +
  scale_color_manual(name="Steigungskoeffizient", labels=c("1","0", "0.3","2"), values=c("black", "blue","gree
```

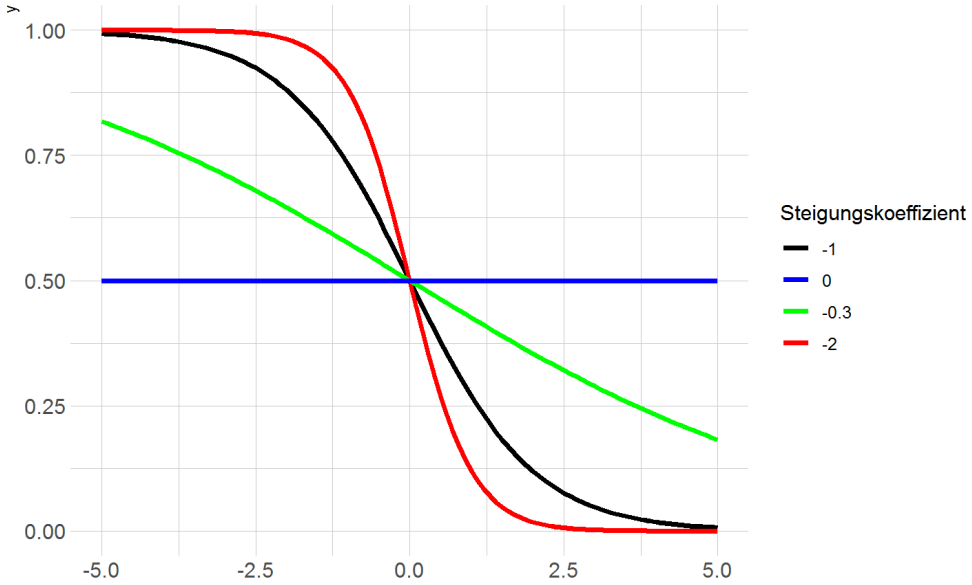


- größerer Steigungskoeffizient: bessere Vorhersage durch Prädiktor
- Steigungskoeffizient = 0, kein Prädiktiver Einfluss

Funktionsverlauf

Steigung

```
ggplot() +
  geom_function(fun = ~exp(-.x)/(1+exp(-.x)), xlim = c(-5,5),      aes(color = 'black'), linewidth = 1.3) +
  geom_function(fun = ~exp(-.x*2)/(1+exp(-.x*2)), xlim = c(-5,5), aes(color = 'red'  ), linewidth = 1.3) +
  geom_function(fun = ~exp(-.x*0.3)/(1+exp(-.x*0.3)), xlim = c(-5,5), aes(color = 'green'), linewidth = 1.3) +
  geom_function(fun = ~exp(-.x*0)/(1+exp(-.x*0)), xlim = c(-5,5), aes(color = 'blue'), linewidth = 1.3) +
  scale_color_manual(name="Steigungskoeffizient", labels=c("-1","0", "-0.3","-2"), values=c("black", "blue","g
```



Beispiel

Hufeisenkrebse

- Studie zu Partnern von weibl. Hufeisenkrebse
- `df_crabs <- rio::import('https://raw.githubusercontent.com/Statistican/Datasets/main/Crabs.csv')`
- zusätzliche Männchen (*sat*) um Nester von weibl. Hufeisenkrebsen
- Bereiten Sie den Datensatz auf:
  - *color:* Faktor: 1, mittel-hell; 2, mittel; 3, mittel-dunkel; 4, dunkel
  - *spine:* Faktor: 1, beide gut; 2, einer gebrochen; 3, beide gebrochen
  - *width:* Schalendicke in cm
  - *weight:* Gewicht in g
  - *sat:* Anzahl männlicher Satelliten
  - *y:* Vorhandensein männlicher Satelliten
-

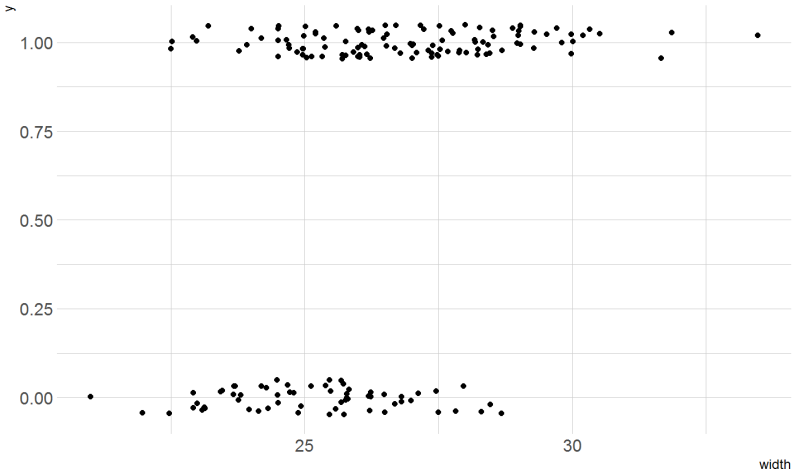
```
df_crabs <- rio::import('https://raw.githubusercontent.com/Statistican/Datasets/main/Crabs.csv')

df_crabs <-
  df_crabs %>%
  mutate(color = factor(color, levels = 1:4,
                        labels = c('mittel-hell' , 'mittel' , 'mittel-dunkel', 'dunkel'))) %>%
  mutate(spine = factor(spine, levels = 1:3,
                        labels = c('beide gut' , 'einer gebrochen' , 'beide gebrochen')))
```

Beispiel

Hufeisenkrebse

- Welche Variablen für einfache logistische Regression?
- AV: *sat*
- UV: *width*
- Visualisieren Sie die beiden Variablen im Streudiagramm.

```
df_crabs %>%
  ggplot(aes(width, y)) +
  geom_jitter(height = 0.05)
```



Logistische Regression

Berechnung

- ähnlicher Aufruf wie bei linearer Regression
- 2 entscheidende Unterschiede
  - statt `lm()` nun `glm()`
  - zusätzliches Argument: `family = binomial`

```
log_reg <- glm(y ~ width, family = binomial, data = df_crabs)
```

Logistische Regression

Berechnung

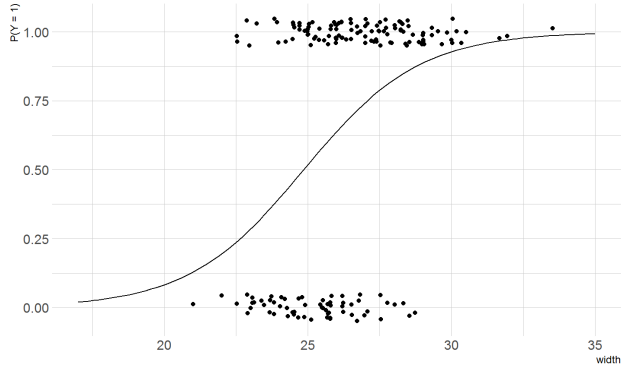
- Inspektion des R-Objekts
- `names(log_reg)`
- `str(log_reg)`
- `summary(log_reg)`
- `broom::tidy(log_reg, conf.int = T)`
- ↯ summary und tidy liefern die logarithmierten Odds Ratios
- `broom::tidy(log_reg, conf.int = T, exponentiate = T)`
- ✓ mit `exponentiate = T` werden die Koeffizienten exponentiert und werden direkt als Odds Ratios berichtet

Regressionsgleichung

- durch log. Regression ermittelte Koeffizienten:
- $\text{logit}(Y = 1|X) = -12.35 + 0.497 \times X$
- geschätzte Wahrscheinlichkeit:
- $\hat{\pi}(x) = \frac{\exp(-12.35+0.497\times X)}{1+\exp(-12.35+0.497\times X)}$

- graphisch:

```
df_crabs %>%
  ggplot(aes(width, y)) +
  geom_jitter(height = 0.05) +
  geom_function(fun =
    ~exp(-12.35 + 0.497 * .x)/
      (1+exp(-12.35 + 0.497 * .x))) +
  xlim(17,35) +
  labs(y = "P(Y = 1)")
```



Interpretation

Steigungskoeffizient

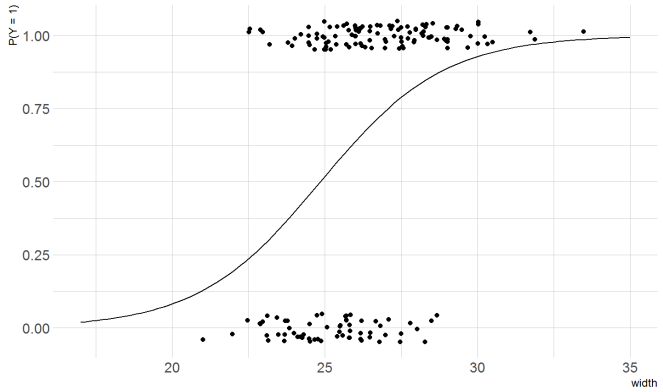
- mit Zunahme von X um 1 Einheit steigt das *logit* um genau β Einheiten
- schwierig interpretierbar
- nach Exponenzierung beider Seiten der logistischen Funktion:
  - $\frac{\pi(x)}{1-\pi(x)} = exp(\alpha + \beta x) = e^{\alpha}(e^{\beta})^{x}$
  - nun werden wieder Odds dargestellt
- neue Interpretationsmöglichkeit:
  - Die Odds an Stelle x ändern sich zu der Stelle x+1, indem diese mit e^{β} multipliziert werden
  - entspricht der Interpretation der Koeffizienten wie in `broom::tidy(log_reg, conf.int = T, exponentiate = T)` berichtet

Interpretation

Steigungskoeffizient

- Bsp.
  - P(x=22.5) = 0.24 (siehe Abbildung)
  - Odds = 0.24/0.76 = 0.316
  - P(x=23.5) = 0.34 (siehe Abbildung)
  - Odds = 0.34/0.66 = 0.515
  - Differenz:
    $0.316 \times exp(0.497) = 0.519$
  - exp(0.497) = 1.64, dh 64% Erhöhung bei 1 zusätzlichen cm

```
df_crabs %>%
  ggplot(aes(width, y)) +
  geom_jitter(height = 0.05) +
  geom_function(fun =
    ~exp(-12.35 + 0.497 * .x)/
      (1+exp(-12.35 + 0.497 * .x))) +
  xlim(17,35) +
  labs(y = "P(Y = 1)")
```



Interpretation

Steigungskoeffizient

- unter Berücksichtigung des KI

```
broom::tidy(log_reg, conf.int = T, exponentiate = T) %>% flex()
```

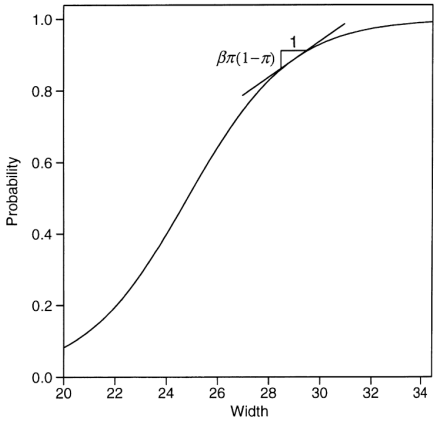
term	estimate	std.error	statistic	p.value	conf.low	conf.high
(Intercept)	0.00	2.63	-4.70	p < .001	0.00	0.00
width	1.64	0.10	4.89	p < .001	1.36	2.03

- eine Zunahme der Größe um 1 cm erhöht die Odds für das Vorhandensein eines Satelliten um mindestens 36% und um maximal 103% (Verdopplung).

Interpretation

Steigungskoeffizient

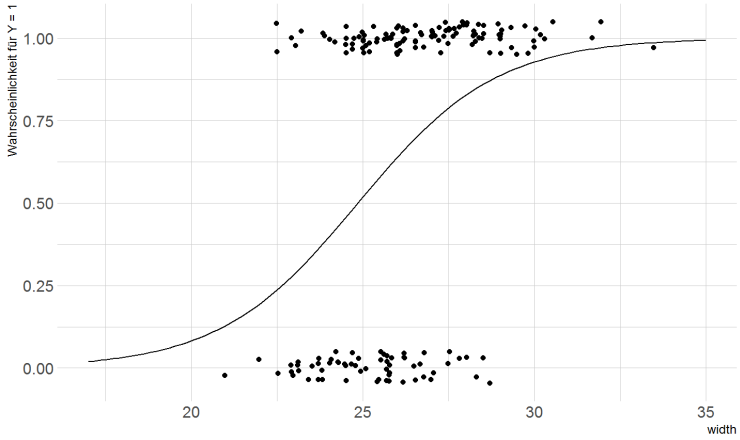
- weitere Interpretation: lineare Approximation
  - direkt für modellierte Wahrscheinlichkeiten
  - die logistische Funktion ist gekrümmt
  - die Auswirkung der Änderung in x hängen entsprechend vom x-Wert ab
  - Tangente an Stelle x hat folgenden Antieg:
  - $\beta \times \pi(x) \times [1 - \pi(x)]$
- im Beispiel:
  - P(x=22.5) = 0.24
  - $0.497 \times 0.24 \times 0.76 = 0.09$
  - P(x=22.5 + 1 = 23.5) = 0.24 + 0.09 = 0.33



Interpretation

Steigungskoeffizient

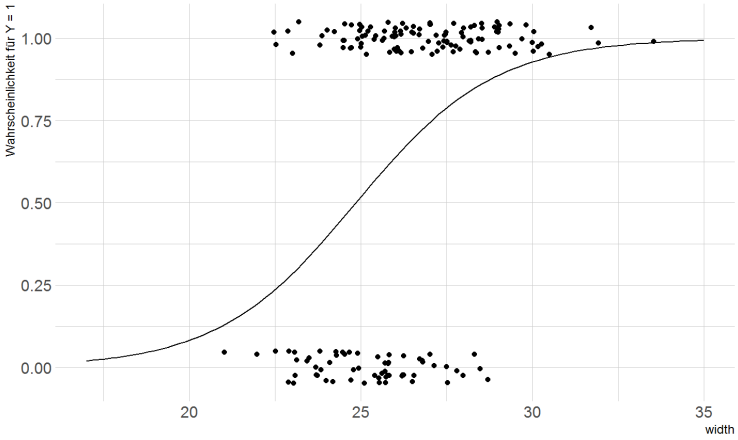
- lineare Approximation:
  - Steigung geht gegen 0, wenn $\pi(x) \to 0$ und $\pi(x) \to 1$
  - maximale Tangentensteigung bei $\pi(x) = 0.50$
  - $\beta \times 0.5 \times 0.5 = 0.25 \times \beta$
  - Bestimmung des x-Wertes an genau dieser Stelle: $x = \frac{-\alpha}{\beta}$
  - = *median effective level*



Interpretation

Steigungskoeffizient

- maximale Tangentensteigung im Krabben-Beispiel
  - 0.497 * 0.50 * 0.50 = 0.124
- median effective level:
  - (-(-12.35))/0.4972 = 24.8391
- Bei einer Größe von 24.8 cm ist die Wahrscheinlichkeitsänderung mit 12.4% pro 1 cm Unterschied maximal



Prädiktion

- Vorhergesagter Wert für best. x-Wert
  - `predict(log_reg, newdata = tibble(width = 22.5))`
  - vorhergesagter Wert auf logit-Skala
  - vom Logit zu Odds: exp(-1.16) = 0.3134862
  - von Odds zur Wk.:
  - 0.3125 / (1+0.3125) = 0.238
  - Ein Krebs mit 22.5 cm hat mit einer Wk. von 23.8% einen Satelliten
- das geht aber auch einfacher, oder?!
- Vorhergesagte Wahrscheinlichkeit für best. x-Wert
  - `predict(log_reg, newdata = tibble(width = 22.5), type = 'response')`
  - $P(Y = 1|X = 22.5) = 0.238$
  - Ein Krebs mit 22.5 cm hat mit einer Wk. von 23.8% einen Satelliten
  - noch besser: `augment(log_reg, newdata = tibble(width = 22.5), type.predict = 'response')`
- ⚠ Das `type`-Argument bei predict-Funktion beachten!

Prädiktion

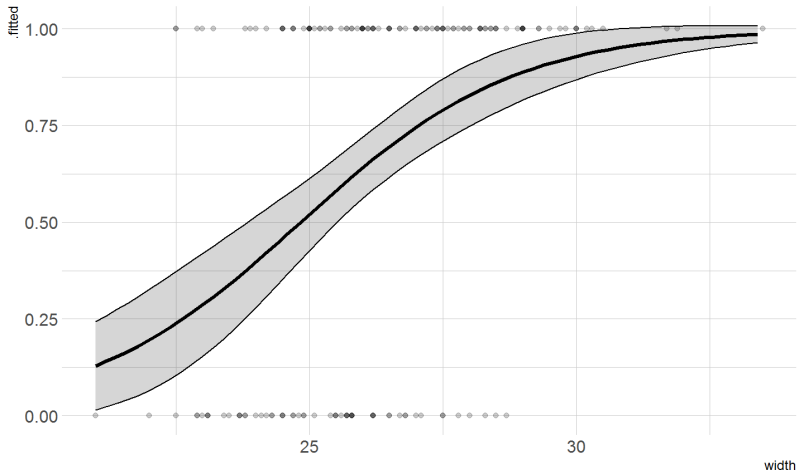
- Mit welcher Wahrscheinlichkeit hat die kleinste Krabbe einen Satelliten?
  - $P(Y = 1 \mid x = 21.0) = 0.129$
- Und die größte Krabbe?
  - $P(Y = 1 \mid x = 33.5) = 0.987$
- Mit welcher Wk. haben durchschnittlich große Krabben einen Satelliten?
  - $P(Y = 1 \mid x = 26.3) = 0.674$
- Wie groß ist die Tangentensteigung an dieser Stelle? Was bedeutet das?
  - $0.497 * 0.674 * 0.326 = 0.11$
  - nahe des Durchschnitts bedeutet eine 1 cm-Änderung eine Veränderung von 11%-Punkten

Prädiktion

Visualisierung

- Visualisierung der Vorhersage, inkl. 95%-KI:

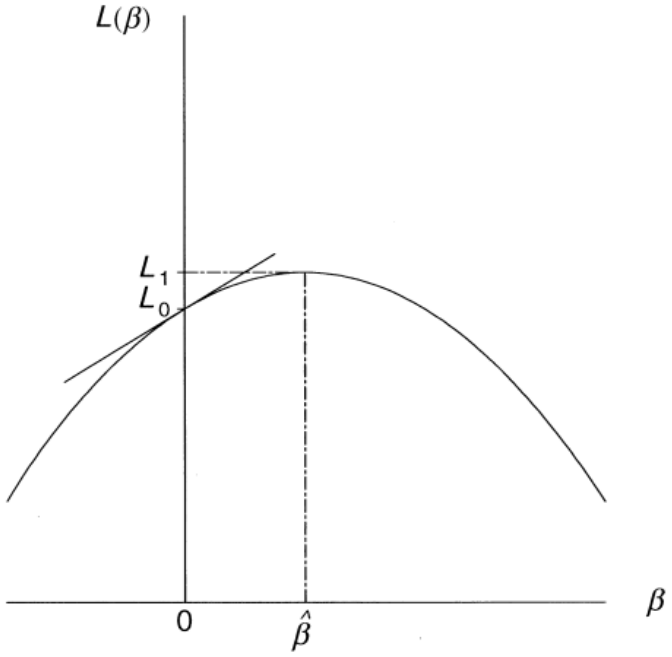
```
min_val <- min(df_crabs$width)
max_val <- max(df_crabs$width)
range_val <- seq(from = min_val, to = max_val, by = .2)

augment(log_reg, newdata = tibble(width = range_val), se_fit = T, type.predict = 'response') %>%
  mutate(ci_ll = .fitted - 1.96 * .se.fit) %>%
  mutate(ci_ul = .fitted + 1.96 * .se.fit) %>%
  ggplot(aes(x=width, y = .fitted, ymin = ci_ll, ymax = ci_ul)) +
  geom_ribbon(alpha = 0.2, color = 'black') + geom_line(linewidth = 1.2) +
  geom_point(aes(width, y), data = df_crabs, inherit.aes = F, alpha = 0.2)
```



Logistische Regression

Koeffizientenschätzung

- OLS: kleinste-Quadrate-Methode
- Maximum-Likelihood-Schätzung:
- die Beta-Koeffizenten werden so geschätzt, dass folgende Funktion maximal wird:
- $l(\beta_0, \beta_1) = \prod_{i:y_i=1} p(x_i) \prod_{i':y_{i'}=0} (1 - p(x_i'))$
- schematische Darstellung der Log-Likelihood-Funktion:



Koeffizientenprüfung

Wald-Test

- Prinzipiell auch wieder mit ANOVA:
  - Wald-Statistik: $\hat{\beta} / SE(\beta)$
  - `tidy(log_reg)`
  - $z = 0.497 / 0.102 = 4.89$
  - $z^2 = 4.89^2 = 28.88$
  - $z^2 \sim \chi(df = 1)$
  - `pchisq(23.88, 1, lower = F)`
- in R:

```
car::Anova(log_reg, test = 'Wald') %>% rownames_to_column('term') %>% flex()
```

term	Df	Chisq	Pr(>Chisq)
width	1	23.89	p < .001

Koeffizientenprüfung

LR-Test

- höhere Power hat die Likelihood-Ratio-Statistik (LR-Test)
- insbesondere auch bei kleinen Stichprobenumfängen
  - $2(L_1 - L_0)$
  - L_0: Funktionswert bei $\beta = 0$ (Nullmodell)
  - L_1: Funktionswert bei beliebigem β (unrestricted Modell)
  - `glance(log_reg)`
  - $225.76 - 194.45 = 31.31$
  - $df = 172 - 171 = 1$
  - folgt ebenfalls Chi²-Verteilung
  - `pchisq(31.31, 1, lower = F)`
- in R:

```
car::Anova(log_reg, test = 'LR') %>% rownames_to_column('term') %>% flex()
```

term	LR Chisq	Df	Pr(>Chisq)
width	31.31	1	p < .001

Koeffizientenprüfung

LR-Test

- Gesamtmodell
- wieder im direkten Vergleich mit Nullmodell
- Nullmodell:

```
log_reg_null <- glm(y ~ 1, family = binomial, data = df_crabs)
```

- Modellvergleich

```
anova(log_reg_null, log_reg, test = 'LR') %>% rownames_to_column('Modell') %>% flex()
```

Modell	Resid. Df	Resid. Dev	Df	Deviance	Pr(>Chi)
1	172	225.76			
2	171	194.45	1	31.31	p < .001

- da nur ein Prädiktor im Modell, entsprechend des Sig-Tests zum Einzelkoeffizienten

Residuen

einfach

- **einfache Residuen**
- einfache Abweichung zwischen bebachtetem und vorhergesagtem Wert
- immer im Berich $[-1; 1]$
- Formel: $\epsilon_i = y_i - \hat{\pi}_i$
- `augment(log_reg, type.predict = 'response') %>% head(1)`
- $\epsilon_i = 1 - 0.848 = 0.152$
- direkt: `residuals(log_reg, type = "response")`

Residuen

Pearson

- Normierung des rohen/einfachen Residuums durch Division mit seiner Standardabweichung
- behebt Varianzinhomogenität (vgl. 0.01*0.99 mit 0.50*0.50)
- Formel: $r_i = \frac{\epsilon_i}{\sqrt{\hat{\pi}_i(1-\hat{\pi}_i)}}$
- `0.152 / sqrt(0.848*(1-0.848)) = 0.423`
- direkt: `residuals(log_reg, type = "pearson")`

Residuen

standardisiert Pearson

- adjustiert Pearson-Residuum um seine Hebewirkung
- $rs_i = \frac{r_i}{\sqrt{1-\hat{h}_i}}$
- $\hat{h}_i$ ... `.hat` in `augment()`
- `0.423 /(sqrt(1-0.0123)) = 0.4256`
- direkt: `rstandard(log_reg, type='pearson')`
- folgen bei Gültigkeit des Modells einer Standardnormalverteilung
- eignen sich für Residuendiagnostik

Residuen

Devianz

- **Devianz-Residuen**
- `augment(log_reg, type.predict = 'response', type.residuals = 'deviance')`
- Fall 1: $\hat{\pi} = 0.848$ und $Y = 1$
- $e_i = \sqrt{-2 \times \log(|\hat{\pi} - (1 - y_i)|)}$
- `sqrt( -2 * log (0.848-(1-1)) ) = 0.574` (zusätzlich gleiches Vorzeichen wie Residuum)
- bei perfektem Fit: $\hat{\pi} = 1.0$ bzw. $\hat{\pi} = 0.0$, log(1) = 0
- bei schlechtem Fit: log(0.001) -> zunehmende Devianz
- direkt: `residuals(log_reg)`
- direkt: `augment(log_reg)$.resid`
- Summe der quadrierten Devianz Residuen = **Devianz des Modells**
- `augment(log_reg)$.resid^2 %>% sum()`
- `deviance(log_reg)`
- `glance(log_reg)$deviance`

Residuen

- Welche Beobachtungen (Krabben) haben die größten / kleinsten Residuen.
  Haben diese Satelliten? Was schätzt das Modell für eine Wahrscheinlichkeit?

Ausreißerdiagnostik

einflussreiche Fälle

- analog zur OLS
- standardisierte Residuen
- Cooks Distanz

Kategoriale Prädiktoren

- analog zum OLS auch hier kategoriale / qualitative Merkmale als Prädiktoren möglich
- Berücksichtigung als Dummy-Variablen / Indikator-Variablen
- Für 2 binäre Variablen X und Z:
  - jeweils 0/1 kodiert
  - $\text{logit}[P(Y=1)] = \beta_0 + \beta_1 x + \beta_2 z$
- alle möglichen Werte-Kombinationen beschreibbar:
- Logits der 4 Kombinationen:

x	z	Logit
0	0	β_0
1	0	$\beta_0 + \beta_1$
0	1	$\beta_0 + \beta_2$
1	1	$\beta_0 + \beta_1 + \beta_2$

Kategoriale Prädiktoren

Beispiel

- Studie zu Cannabiskonsum
- Faktoren, die Cannabiskonsum beeinflussen
- Laden Sie den Datensatz: `df_canna <- rio::import("https://github.com/Statistican/Datasets/raw/main/Cannabiskonsum.Rdata", trust = T)`
- ✎ Fitten Sie ein Logistisches Regressionsmodell zur Prädiktion von Cannabiskonsum. Verwenden Sie beide Prädiktoren. Lassen Sie sich die Koeffizenten anzeigen.
  - Geschlecht (Ref = 'männlich'): $\beta = 0.203$; $e^{\beta} = 1.22$
  - Ethnie (Ref = 'Andere'): $\beta = 0.444$; $e^{\beta} = 1.56$
  - d.h. für beide Ethnien haben Männer 1.22-fach höhere Odds als Frauen Cannabis zu konsumieren.

Kategoriale Prädiktoren

Beispiel

- ✎ Berechnen Sie doch dazu auch nochmal das Odds Ratio zu Cannabiskonsum in Abhängigkeit von Geschlecht per Hand!
- Grundlage: 4 Felder-Tafel

```
table(df_canna$Geschlecht, df_canna$Mariuhana)
```

```
##
##              0   1
##   weiblich 675 445
##   männlich 641 515
```

- Quoten berechnen:
  - Frauen: 445/675 = 0.659
  - Männer: 515/641 = 0.803
  - 0.803/0.659 = 1.22

Kategoriale Prädiktoren

Referenzkategorie

- ✎ Was passiert wenn die Referenzkategorie bei Geschlecht getauscht wird (auf männlich). (`rstatix::set_ref_level()`)

- Ref: weiblich

term	estimate	std.error	statistic	p.value
(Intercept)	-0.83	0.17	-4.93	p < .001
Geschlechtmännlich	0.20	0.09	2.38	p = 0.017
Ethnieweiß	0.44	0.17	2.65	p = 0.008

- Ref: männlich

term	estimate	std.error	statistic	p.value
(Intercept)	-0.63	0.17	-3.78	p < .001
Geschlechtweiblich	-0.20	0.09	-2.38	p = 0.017
Ethnieweiß	0.44	0.17	2.65	p = 0.008

- die Koeffizienten (log(OR)) tauschen das Vorzeichen
- die exponenzierten Koeffizienten (ORs) sind invertiert
- der Intercept ändert sich entsprechend
- die p-Werte sind unverändert!

Multiple log. Regression

- wie bei OLS-Regression Erweiterung um Prädiktoren möglich
- kontinuierliche und/oder kategoriale Prädiktoren
- Formeln erweitern sich logisch:
  - $P(Y = 1) = \pi = \frac{\exp(\alpha + \beta_1 x_1 + \beta_2 x_2 + \beta_3 x_3)}{1 + \exp(\alpha + \beta_1 x_1 + \beta_2 x_2 + \beta_3 x_3)}$
  - $\log\left[\frac{\pi(x)}{1-\pi(x)}\right] = \alpha + \beta_1 X_1 + \beta_2 X_2 + \beta_3 X_3$

Beispiel

Hufeisenkrebse

- Erweiterung des Modells zu den Hufeisenkrebsen
- zur Schalendicke (*width*) zusätzlich Farbe (*color*) als Prädiktor
- Aufruf in R

```
log_reg2 <- glm(y ~ width + color, family = binomial, data = df_crabs)
```

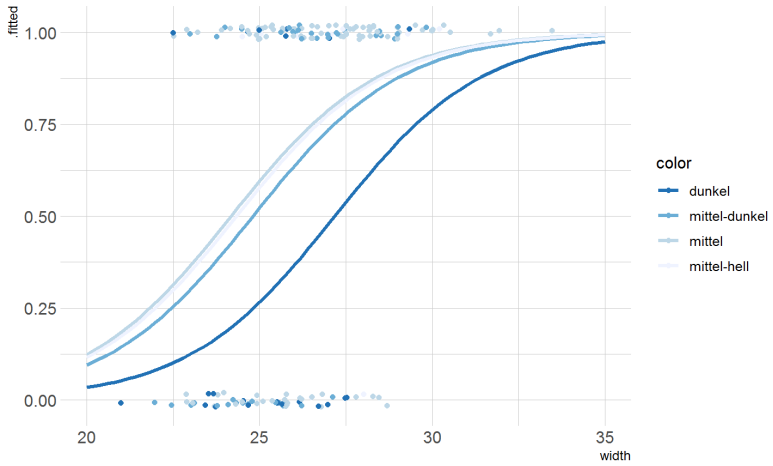
Beispiel

Hufeisenkrebse

- Koeffizienten inspizieren:

```
tidy(log_reg2, exp = T) %>% flex()
```

term	estimate	std.error	statistic	p.value
(Intercept)	0.00	2.87	-3.96	p < .001
width	1.60	0.11	4.43	p < .001
colormittel	1.08	0.74	0.10	p = 0.922
colormittel-dunkel	0.80	0.78	-0.29	p = 0.773
colordunkel	0.26	0.85	-1.56	p = 0.119



- Farbe “mittel-hell” bildet Referenzkategorie
- Modell ohne Interaktion -> parallele Linien
- P(width = 26.3, Farbe = dunkel) = 0.40
- Odds: 0.4/0.6 = 0.67
- P(width = 26.3, Farbe = mittel-hell) = 0.72
- Odds: 0.72/0.28 = 2.57
- Odds Ratio = 0.67/2.57 = 0.26
- entspricht dem Koeffizenten für dunkle Krabben (OR = 0.26)

Modellvergleich

- aber ist das Modell mit Farbe als Prädiktor besser?
- bei OLS: Vergleich der Fehlerquadratsummen ($SS_{Residuum}$) mittels ANOVA
- für log. Regressionsmodelle: Vergleich der Devianz der Modelle
- **Likelihood-Ratio-Test**
  - Vgl. der maximierten Log-Likelihoods
  - L_0: `logLik (log_reg)` (nur *width*): -97.23 (df=2)
  - L_1: `logLik (log_reg2)` (mit *color*): -93.73 (df=5)
  - $2(L_1 - L_0)$: 2*(-93.73-(-97.23)) = 7.00
  - entspricht der Differenz der Devianzen der Modelle (vgl. `glance()`)
  - Chi²-Verteilt mit df=(L_1-L_0): 5-2 = 3
- in R:

term	df.residual	residual.deviance	df	deviance	p.value
y ~ width	171	194.45			
y ~ width + color	168	187.46	3	7	p = 0.072

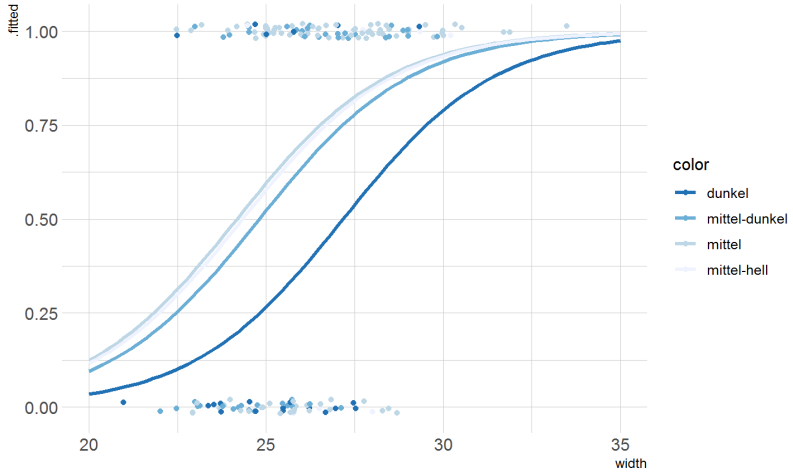
Modellvergleich

- 2 verschiedene ANOVAS!
  - `stats::anova(test = "LRT")`: Vergleich zwischen Modellen
  - `car::Anova(test = "LR")`: Vergleich eines Modells gegen Nullmodell, Koeffizientenprüfung
- bei `stats::anova`: nutzt Typ 1
- bei `car::Anova`: nutzt Typ 2
- ✎ Vgl:
  - `stats::anova(log_reg_null, log_reg, log_reg2, test ='LRT')`
  - `car::Anova(log_reg_null, log_reg, log_reg2, test ='LR')`
  - mehrere Modelle bei `car::Anova` nicht möglich!
- ✎ Vgl:
  - `stats::anova(log_reg2, test ='LRT')`
  - `car::Anova(log_reg2, test ='LR')`
  - Devianz des *width*-Effektes bei `stats::anova` überhöht

Hufeisenkrebse

Sparsamkeit

- kategorialer Prädiktor “verbraucht” 3 Freiheitsgrade
- prinzipiell sind sparsame Modelle zu bevorzugen
  - weniger Parameter
  - mehr statistische Power
  - bessere Interpretierbarkeit
  - höhere Generalisierbarkeit
- sind die Effekte unterschiedlicher Farben sehr unterschiedlich?



- die drei helleren Farben haben fast den gleichen Effekt (gleiche geschätzte Wk.)

Hufeisenkrebse

Zusammenfassung Farben

- Berechnen sie noch ein drittes Modell (log_reg3), bei dem die drei helleren Farbkategorien in eine einzige Kategorie zusammengefasst werden. Überprüfen Sie ob sich dieses Modell signifikant von den beiden anderen Modellen (nur *width* bzw. *width* und *color* als Faktor) unterscheidet. Welches Modell wäre dann zu bevorzugen.
- Umkodieren:

```
# umkodieren Option 1
df_crabs <-
  df_crabs %>%
  mutate(color_dichotom = fct_collapse(color,
    hell = c('mittel-hell', 'mittel', 'mittel-dunkel'),
    dunkel = c('dunkel')
    ))

# umkodieren Option 2
df_crabs <-
  df_crabs %>%
  mutate(color_dichotom = ifelse(color == 'dunkel', 'dunkel', 'hell') %>% as.factor())
```

Hufeisenkrebse

Zusammenfassung Farben

- Berechnen

```
log_reg3 <- glm(y ~ width + color_dichotom, family = binomial, data = df_crabs)
```

- Modellvergleich 1:

```
anova(log_reg,  log_reg3, test = 'LR') %>% tidy() %>% flex()
```

term	df.residual	residual.deviance	df	deviance	p.value
y ~ width	171	194.45			
y ~ width + color_dichotom	170	187.96	1	6.49	p = 0.011

- Modellvergleich 1:

```
anova(log_reg3, log_reg2, test = 'LR') %>% tidy() %>% flex()
```

term	df.residual	residual.deviance	df	deviance	p.value
y ~ width + color_dichotom	170	187.96			
y ~ width + color	168	187.46	2	0.5	p = 0.778

Modellgüte

- Vergleich des Modells mit dem Nullmodell (LR-Test)
- Hosmer-Lemeshow-Test (Kalibrierung)
- Kalibrierungsplot
- Pseudo-R^2
- (Klassifikationsergebnisse)
- (AUC, c-Statistik)

Modellgüte

H-L-Test

- Kalibrierung:
  - vorhergesagte Wk. einer Gruppe von Fällen entspricht tatsächlichem Anteil am Kriteriumswert
- **Hosmer-Lemeshow-Test**
  - Test zur Anpassungsgüte des Modells
  - H_0 **:** Der Anteil beobachteter und erwarteter Erfolge ist gleich
- praktisch:
  - 10 Dezile der gefitteten Wahscheinlichkeiten
  - Anteil der beobachteten Erfolge je Kategorie
  - Vergleich der beobachteten und erwarteten Erfolge

Modellgüte

H-L-Test

- Teststatistik in R:

```
performance::performance_hosmer(log_reg3) %>%
  as.data.frame() %>%
  flex()
```

chisq	df	p.value
4.02	8	p = 0.855

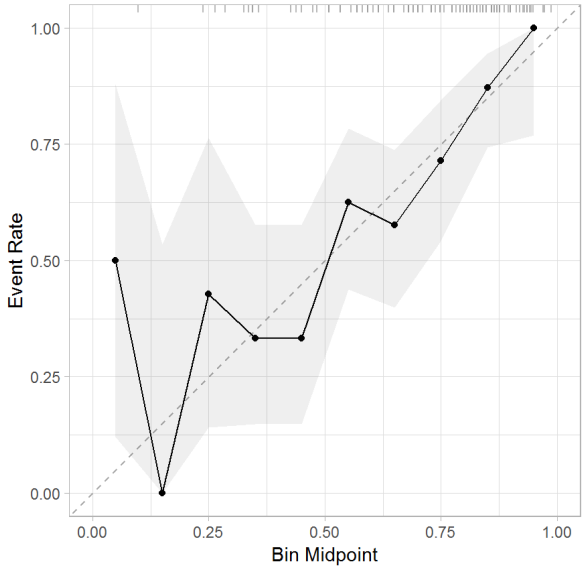
- Interpretation:
  - nicht sig: gute Modellanpassung
  - tw. zu sensitiv
  - varlässlich erst ab $p < \frac{n}{10}$

Modellgüte

Kalibrierungsplot

- Kalibrierungsplot:

```
augment(log_reg3, type.predict = 'response') %>%
  probably::cal_plot_breaks(y, .fitted, num_breaks = 10)
```



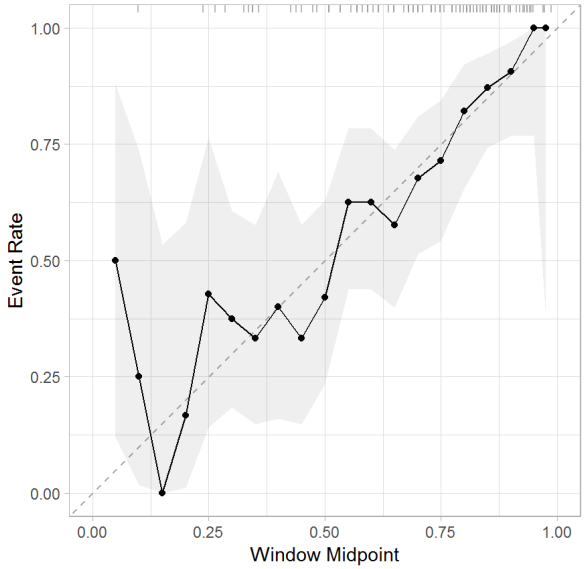
- im Ideallfall alle Punkte auf Diagonale
- ⚠ Anghängigkeit von bins!
- ✎ Unterschiedliche `num_breaks` testen.

Modellgüte

Kalibrierungsplot

- überlappende “Windows”

```
augment(log_reg3, type.predict = 'response') %>%
  probably::cal_plot_windowed(y, .fitted)
```

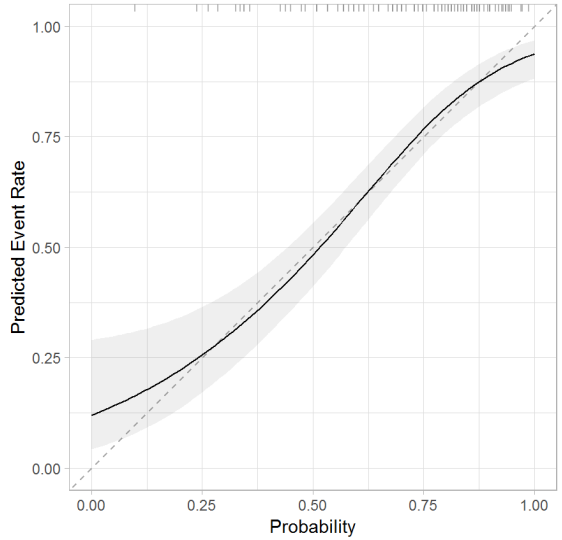


Modellgüte

Kalibrierungsplot

- modelliert:
  - intern: logistisches Modell
  - Vorhersage von Y durch . $fitted$

```
augment(log_reg3, type.predict = 'response') %>%
  probably::cal_plot_logistic(y, .fitted)
```



  - keine Annahmen notwendig

Modellgüte

Pseudo-R^2

- keine 1:1 Entsprechung des OLS-R^2
- für Maximum-Likelihood basierte Modelle
- idR auch im Intervall $[0; 1]$
- verschiedene Interpretationsansätze:
  - erklärte Varianz
  - Verbesserung ggü Nullmodell
  - Quadrat der Korrelation
- verschiedenste Maße:
  - Cox & Snells R^2
  - Nagelkerkes R^2
  - McFaddens R^2
  - Tjurs R^2
  - Efrons R^2

Modellgüte

Cox & Snells R²

- **Cox & Snells R²**
  - Grad der Verbesserung des vollständigen Modells mit Prädiktoren gegenüber dem Nullmodell
  - Ratio der Likelihoods der Modelle
  - Formel: $\mathrm{R}^2_{\mathrm{Cox\&Snell}} = 1 - \left(\frac{L_0}{L_1}\right)^{2/n}$
  - Bereich: $[0; 1)$, 1 wird nicht erreicht
- ✎ Berechnen Sie Cox & Snells R² für das Modell *log_reg3* von Hand!
- Lösung:
  - `1-(exp(logLik(log_reg_null))/exp(logLik(log_reg3)))^(2/173)`
  - `1-(exp(-112.8793)/exp(-93.97894))^(2/173) = 0.196`
- in R: `performance::r2_coxsnell(log_reg3)`

Modellgüte

Nagelkerkes R²

- **Nagelkerkes R²**
  - Modifikation von Cox & Snells R²
  - Resaklierung, das 1 als Maximalwert erreicht werden kann
  - Formel: $\mathrm{R}^2_{\mathrm{Cox\&Snell}} = \frac{1-\left(\frac{L_0}{L_1}\right)^{2/n}}{1-L_0^{2/n}}$
  - Bereich: $[0; 1]$
  - gute Anpassung bereits im Bereich $[0.2; 0.4]$
- ✎ Berechnen Sie Nagelkerkes R² für das Modell *log_reg3* von Hand!
- Lösung:
  - `1-(exp(logLik(log_reg_null))/exp(logLik(log_reg3)))^(2/173) / (1-exp(logLik(log_reg_null))^(2/173))`
  - `0.167/0.729 = 0.269`
- in R: `performance::r2_nagelkerke(log_reg3)`

Modellgüte

McFaddens R²

- **McFaddens R²**
  - Grad der Verbesserung des vollständigen Modells mit Prädiktoren gegenüber dem Nullmodell
  - Ratio der Log-Likelihoods bzw. Devianzen der Modelle
  - Formel: $\mathrm{R}^2_{\mathrm{McFadden}} = 1 - \frac{\ln L_1}{\ln L_0} = 1 - \frac{D_1}{D_0}$
  - Bereich: $[0; 1)$, 1 wird nicht erreicht
  - gute Anpassung bereits im Bereich $[0.2; 0.4]$
- ✎ Berechnen Sie McFaddens R² für das Modell *log_reg3* von Hand!
- Lösung:
  - `1-(logLik(log_reg3)/logLik(log_reg_null))`
  - `1-(-93.978/-112.87) = 0.167`
- in R: `performance::r2_mcfadden(log_reg3)`

Modellgüte

Tjurs R^2

- **Tjurs R^2**
  - vgl. der durchschnittlichen geschätzten Wahrscheinlichkeiten der beiden Klassen
  - Differenz der beiden mittleren vorhergesagten Wk.
  - Formel: $\mathrm{R}^2_{\mathrm{Tjur}} = \frac{1}{n_1} \sum \hat{\pi}(y = 1) - \frac{1}{n_0} \sum \hat{\pi}(y = 0)$
  - 0/1 = keine/perfekte diskriminatorische Leistung
- ✎ Berechnen Sie Tjurs R^2 für das Modell *log_reg3* von Hand!
- Lösung:
  - `pred_logreg %>% group_by(y) %>% get_summary_stats(.fitted)`
  - `0.714-0.512 = 0.202`
- in R: `performance::r2_tjur(log_reg3)`

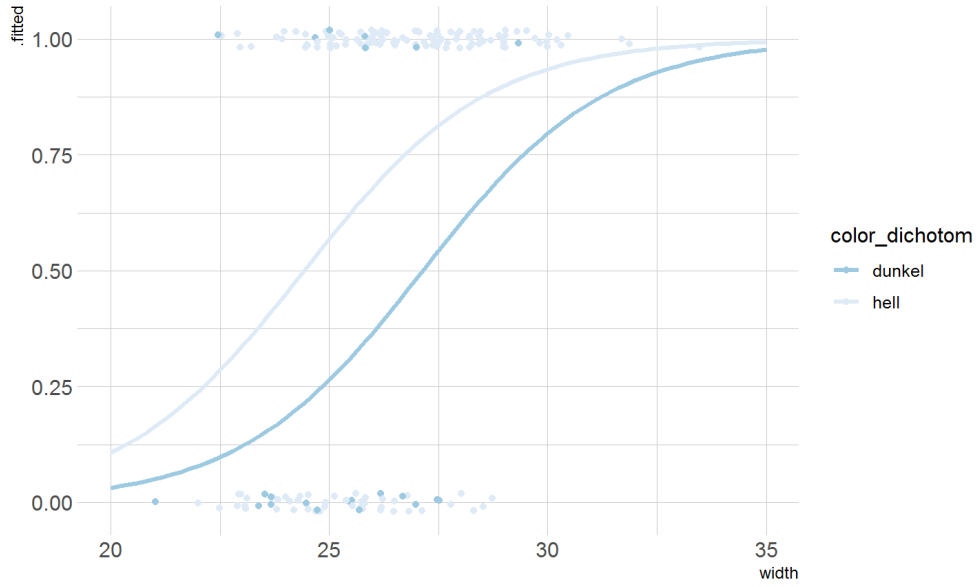
Modellgüte

Efrons R^2

- **Efrons R^2**
  - korrelations-basiertes Maß
  - Bestimmtheitsmaß der OLS am ähnlichsten
  - Interpretation: Quadrat der Korrelation und Anteil erklärter Variabilität
  - Formel: $$\mathrm{R}^2_{\mathrm{Efron}} = 1 - \frac{\frac{1}{n}\sum_{i=1}^{n}(Y_i - \hat{P}_i)^2}{\frac{1}{n}\sum_{i=1}^{n}(Y_i - \bar{Y})^2}$$
  - Bereich: $[0; 1]$
- ✎ Berechnen Sie Efrons R^2 für das Modell *log_reg3* von Hand!
- Lösung:
  - `1/173*sum((pred_logreg$y-pred_logreg$.fitted)^2) = 0.184`
  - `1/173*sum((pred_logreg$y-mean(pred_logreg$y))^2) = 0.230`
  - `1-0.184/0.230 = 0.1997541`
- in R: `performance::r2_efron(log_reg3)`

Interaktion

- ohne Interaktionsterm wird unterstellt, dass der Effekt von *width* für beide Farbkategorien gleich ist
- wie bei der OLS: Linien kreuzen sich nicht / verlaufen parallel
- ggf. sind aber die Effekte von *width* bei Krabben verschiedener Farben unterschiedlich



Interaktion

Hufeisenkrebse

- Berechnen sie noch ein viertes Modell (log_reg4), das eine Interaktion zwischen *width* und der dichotomen Farbvariable modelliert. Überprüfen Sie ob sich dieses Modell signifikant von dem Modell *log_reg3* unterschiedet.

```
log_reg4 <- glm(y ~ width * color_dichotom, family = binomial, data = df_crabs)
anova(log_reg3, log_reg4, test = "LR")
```

- Effekte:

```
tidy(log_reg4) %>% flex()
```

term	estimate	std.error	statistic	p.value
(Intercept)	-5.85	6.69	-0.87	p = 0.382
width	0.20	0.26	0.77	p = 0.444
color_dichotomhell	-6.96	7.32	-0.95	p = 0.342
width:color_dichotomhell	0.32	0.29	1.13	p = 0.260

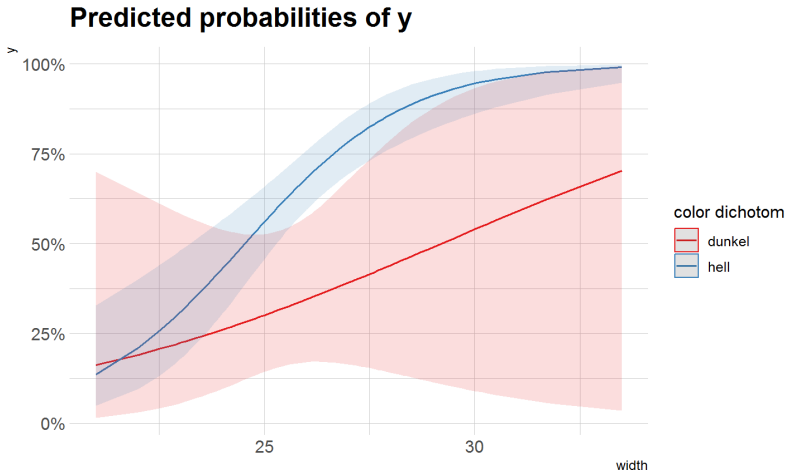
  - helle Krabben: $logit[P(Y = 1)] = -12.81 + 0.52x$
  - dunkle Krabben: $logit[P(Y = 1)] = -5.85 + 0.20x$

Interaktion

Hufeisenkrebse

- nun kreuzen sich die beiden Kurven:

```
sjPlot::plot_model(log_reg4, type = 'pred', terms = c('width [all]', 'color_dichotom'))
```



- an der Stelle:
  - Gleichsetzen der beiden Regressionsgleichungen von oben
  - $-12.81 + 0.52x = -5.85 + 0.20x$
  - $x = 21.6$
  - d.h. hellere Krabben haben fast über die ganze *width*-Breite eine höhere Wk.

Ergenisbericht

- analog zur OLS:
- Koeffiziententabelle:
  - `gtsummary::tbl_regression(log_reg3)`
  - aber mit exonenzierten Werten
  - `gtsummary::tbl_regression(log_reg3, exponentiate = T)`
- Modellgütemaße:
  - `gtsummary::add_glance_source_note()`
- Plots mit vorhergesagten Werten (Wahrscheinlichkeiten)
  - `jtools::effect_plot(log_reg3, pred = 'width', plot.points = T, interval = T)`
  - `jtools::effect_plot(log_reg3, pred = 'color_dichotom', plot.points = T, interval = T)`
  - `sjPlot::plot_model(log_reg3, type = 'pred', terms = c('width', 'color_dichotom'))`
  - `sjPlot::plot_model(log_reg3, type = 'pred', terms = c('color_dichotom', 'width'))`

Ergenisbericht

Prädiktive Leistung

- ϟ Aber was ist mit der Vorhersage der Klassenzugehörigkeit?!
- Vorhersage: `augment(log_reg3, type.predict = 'response')`
- Dichotomisierung der Wk am Wert 0.50!
- in R:

```
pred_logreg <-
  augment(log_reg3, type.predict = 'response') %>%
  mutate(pred_class = ifelse(.fitted > 0.5, 1, 0), .after = ".fitted")
```

- Wie gut stimmt Vorhersage mit der Wirklichkeit überein?

Ergenisbericht

Klassifikationstabelle

- **Klassifikationstabelle**
  - Kreuzklassifikation zwischen Wahrheit (beobachteter Klasse) und vorhergesagter Klasse:

```
pred_logreg %>%
  tbl_cross(pred_class, y) %>%
  flex()
```

	y		
	0	1	Total
pred_class			
0	29	14	43
1	33	97	130
Total	62	111	173

```
# schlichter:
table(pred_logreg$pred_class, pred_logreg$y)
```

```
##
##       0  1
##   0 29 14
##   1 33 97
```

Ergenisbericht

Klassifikationstabelle

- in R:

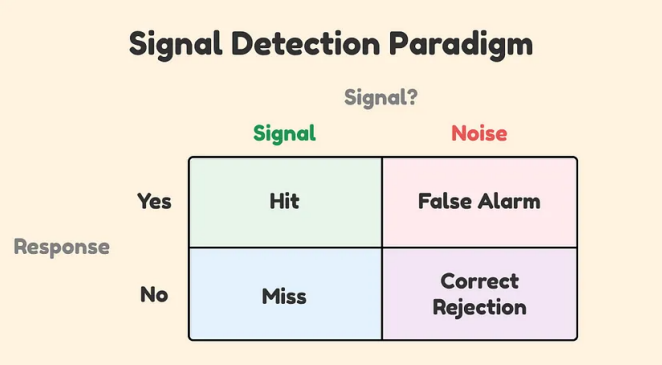
```
caret::confusionMatrix(
  reference = factor(pred_logreg$y),
  data      = factor(pred_logreg$pred_class),
  positive  = "1")
```

- zur Bedeutung der Werte …

Exkurs

Signalentdeckungstheorie

- Bedeutung der Zellen in der 4-Felder-Tafel



	y		
	0	1	Total
pred_class			
0	29	14	43
1	33	97	130
Total	62	111	173

- Beschreibung: `?caret::confusionMatrix()`

Ergenisbericht

Klassifikationstabelle

- der Grenzwert von 0.50 als cut-off zur Vorhersage ist uU problematisch
- zB. bei ungleicher Klassenhäufigkeit
- ggf. verbesserte Vorhersage bei Prävalenzwert
- `mean(df_crabs$y)` = 0.642
- in R:

```
pred_logreg <-
  augment(log_reg3, type.predict = 'response') %>%
  mutate(pred_class2 = ifelse(.fitted > 0.642, 1, 0), .after = ".fitted")

caret::confusionMatrix(
  reference = factor(pred_logreg$y),
  data      = factor(pred_logreg$pred_class2),
  positive  = "1")
```

Ergenisbericht

Klassifikationstabelle

- Bestimmung des optimalen Cut-offs
- Durchprobieren!?
- in R:

```
cut_logreg <- cutpointr::cutpointr(pred_logreg, .fitted, y,
                     method = cutpointr::maximize_metric,
                     metric = cutpointr::sum_sens_spec,
                     pos_class = 1,
                     direction = ">="
                     )
cut_logreg %>%
  select(2:8) %>%
  flex()
```

optimal_cutpoint	method	sum_sens_spec	acc	sensitivity	specificity	AUC
0.67	maximize_metric	1.48	0.72	0.68	0.79	0.77

- ✎ Verwenden Sie den optimalen Cutoff-Wert und führen Sie die obige Klassifikation erneut durch.

Ergenisbericht

Klassifikationstabelle

- Korrektklassifikationsraten von 72.8% bzw. 68.8% sehr optimistisch
- Modell was für alle Krabben den Wert 1 vorhersagen würde hätte einen Wert von 64.2%!
- Einschränkungen der Klassifikationstabelle:
  - Dichotomisierung der kontinuierlichen Wk
  - Wahl des Cut-Offs willkürlich
  - Ergebnisse abh. von rel Häufigkeiten von 0/1-Werten

Ergenisbericht

ROC-Kurven

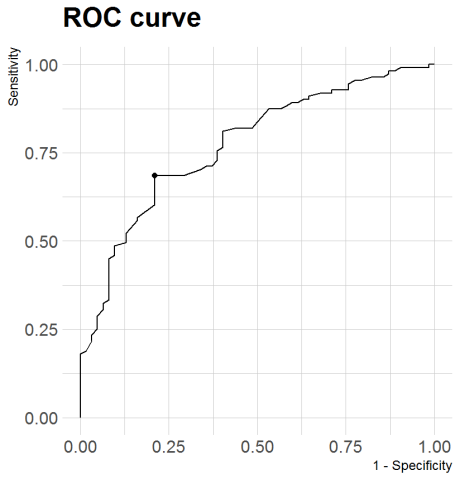
- **Receiver Operating Caracteristic**
  - Plot zur Darstellung von Sensitivität und Spezifität für alle möglichen Cut-Off-Werte
  - x: 1-Spezifität
  - y: Sensitivität

Ergenisbericht

ROC-Kurven

- ROC-Curve

```
cutpointr::plot_roc(cut_logreg)
```



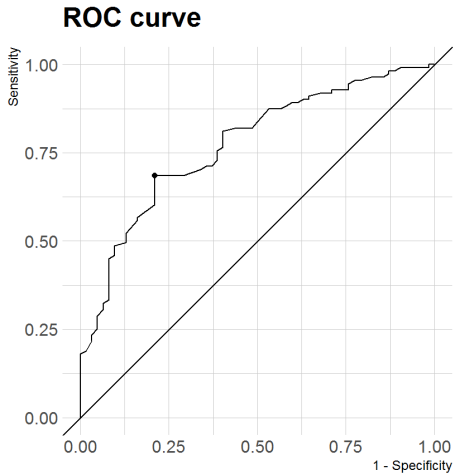
- Diagonale = keine prädiktive Leistung
- je “höher” die Kurve, umso besser die Vorhersagegüte
- Fläche unter der ROC-Kurve als Zusammenfassung

Ergenisbericht

ROC-Kurven

- ROC-Curve

```
cutpointr::plot_roc(cut_logreg) +
  geom_abline(yintercept = 0, slope = 1)
```



- **Area under the Curve** (AUC), c-Statsitik
  - Wert von 0.50 = Zufallsschätzung
  - Wert von 1.00 = perfekte Vorhersage

Ergenisbericht

ROC-Kurven

- Zeichnen Sie die ROC-Kurven für die beiden anderen logistischen Modelle (log_reg und log_reg2). Welche AUC-Werte erreichen diese Modelle?

Stichprobengröße

- mind. N = 50, besser N = 100 (Urban, 1993)
- mind. 25 Fälle je Kriteriumsausprägung (Backhaus, 2008)
- Anzahl der geschätzten Parameter * 10 in am schwächsten besetzter Kriteriumskategorie (Hosmer & Lemeshow, 2000)
- gewisse Mindesthäufigkeiten bei Wertekombinationen
  - erwartete Häufigkeiten in allen Zellen >1 und
  - bei 80% der Zellen >5

Übung 1

Hufeisenkrebse

- Untersuchen Sie den Datensatz zu den Hufeisenkrebsen weiter:
- Lassen sich weitere Prädiktoren identifizieren, die die Vorhersage verbessern?
- Vergleichen Sie die verschiedenen Modelle anhand geeigneter Maße.
- Lassen sich einflussreiche Fälle identifizieren, die die Modellgüte erheblich beeinflussen?
- Visualisieren Sie die Vorhersagen.
- Welche Klassifikationsgüte erreicht das “beste” Modell?

Übung 2

Prostatakrebs

- In einer retrospektiven Studie wurden medizinische und epidemiologische Daten von 244 holländischen Männern im Alter zwischen 55 und 65 Jahren ausgewertet, bei denen aufgrund eines PSA-Wertes (Prostate-Specific Antigen) in der diagnostischen Grauzone von 3-10 g/l ein Verdacht auf Prostatakrebs bestand. Dabei konnten mittels logistischer Regressionsanalyse neben dem Quotienten von freiem (F) und totalem (T) PSA auch ein positiver rektaler Tastbefund (Digital Rectal Examination, DRE) und eine positive Familienanamnese als signifikante diagnostische Marker für Prostatakrebs identifiziert werden.

Parameter	Regressionskoeffizient	95% KI	p-Wert
Achsenabschnitt	-0.169	-.-	-.-
Quotient F :T	-10.197	(-18.932,-3.053)	0.0054
DRE (+)	1.643	(0.752,2.559)	0.0004
Familienanamnese (+)	1.076	(0.012,2.131)	0.0437

1. Wie sind die Resultate der logistischen Regressionsanalyse zu interpretieren? Transformieren Sie dazu die Regressionskoeffizienten der beiden dichotomen Einflussgrößen DRE und Familienanamnese auf geeignete Weise in Odds-Ratios.
2. Ermitteln Sie die Wahrscheinlichkeit für das Vorliegen eines Prostatakrebs bei einem F:T Quotienten von 0.07, positivem Tastbefund und positiver Familienanamnese.
3. Ermitteln Sie die Wahrscheinlichkeit für das Vorliegen eines Prostatakrebs bei einem F:T Quotienten von 0.35, negativem Tastbefund und negativer Familienanamnese. Vergleichen Sie das Ergebnis mit der unter b. ermittelten Wahrscheinlichkeit.

Übung 3

Jahreseinkommen

- Lade Sie den Datensatz, den sie an folgender Stellefinden: https://archive.ics.uci.edu/dataset/2/adult
- Inspizieren Sie die Daten und beschreiben sie die Zusammenhänge graphisch und numerisch.
- Erstellen Sie ein logistischen Regressionsmodell mit dem Sie Vorhersagen ob das Jahreseinkommen einer Person über 50’000 Dollar liegt.
- Wählen Sie geeignete Prädiktoren aus. Prüfen Sie ob Transformationen der Prädiktoren oder Interaktionen die Vorhersage verbessern.
- Welche Vorhersagegüte können Sie erreichen?

Übung 4

Krebsremission

In einer Studie wurden Merkmale untersucht, ob ein Krebspatient eine Remission erreicht (1 = ja, 0 = nein). Eine wichtige erklärende Variable war ein Markierungsindex (LI = Prozentsatz der “markierten” Zellen), der die Proliferationsaktivität von Zellen misst, nachdem ein Patient eine Injektion von tritiiertem Thymidin erhalten hat. Tabelle 4.5 zeigt die Daten und Ergebnisse für ein logistisches Regressionsmodell. (Datensatz: https://raw.githubusercontent.com/Statistican/Datasets/main/Krebsremission.csv)

1. Zeigen Sie, dass P(Y = 1) = 0.50 ist, wenn LI = 26.0.
2. Zeigen Sie, dass sich die geschätzten Odds zur Remission bei einer Erhöhung von LI um 1 um 1.16 ändern.
3. Fassen Sie den LI-Effekt zusammen, indem Sie zeigen, wie sich P(Y = 1) über den Bereich oder den Interquartilsbereich Bereich der LI-Werte.
4. Zeigen Sie, dass die Änderungsrate von P(Y = 1) 0.009 beträgt, wenn LI = 8.