Blatt 1

Vanessa Kleisch

2024-04-18

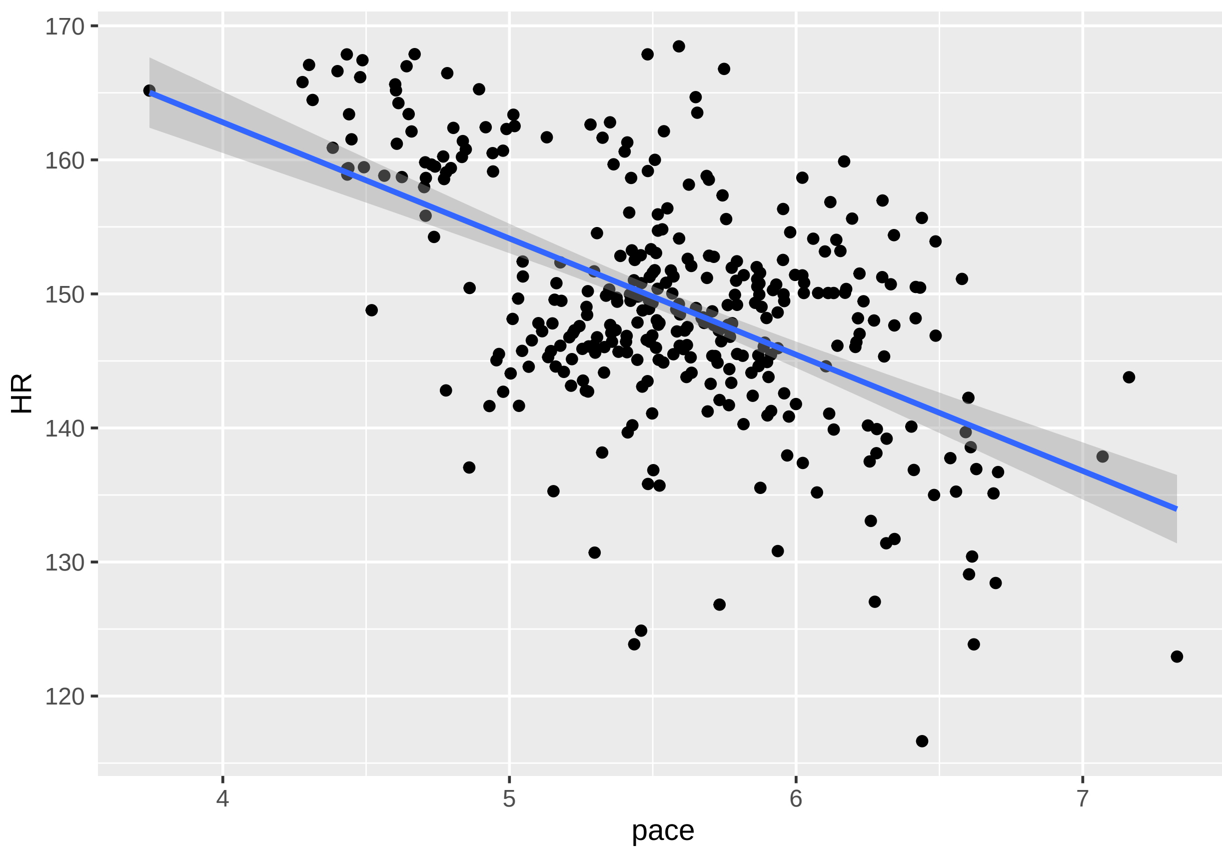
Aufgabe 1

1a

ALternative siehe Lösungsblatt

```
data1 <- read_rds("RunningAgg.Rds")
m1 <- ggplot(data1, aes(x = pace, y = HR)) +
  geom_point()+
  geom_smooth(method = "lm")
m1
```

```
## `geom_smooth()` using formula = 'y ~ x'
```



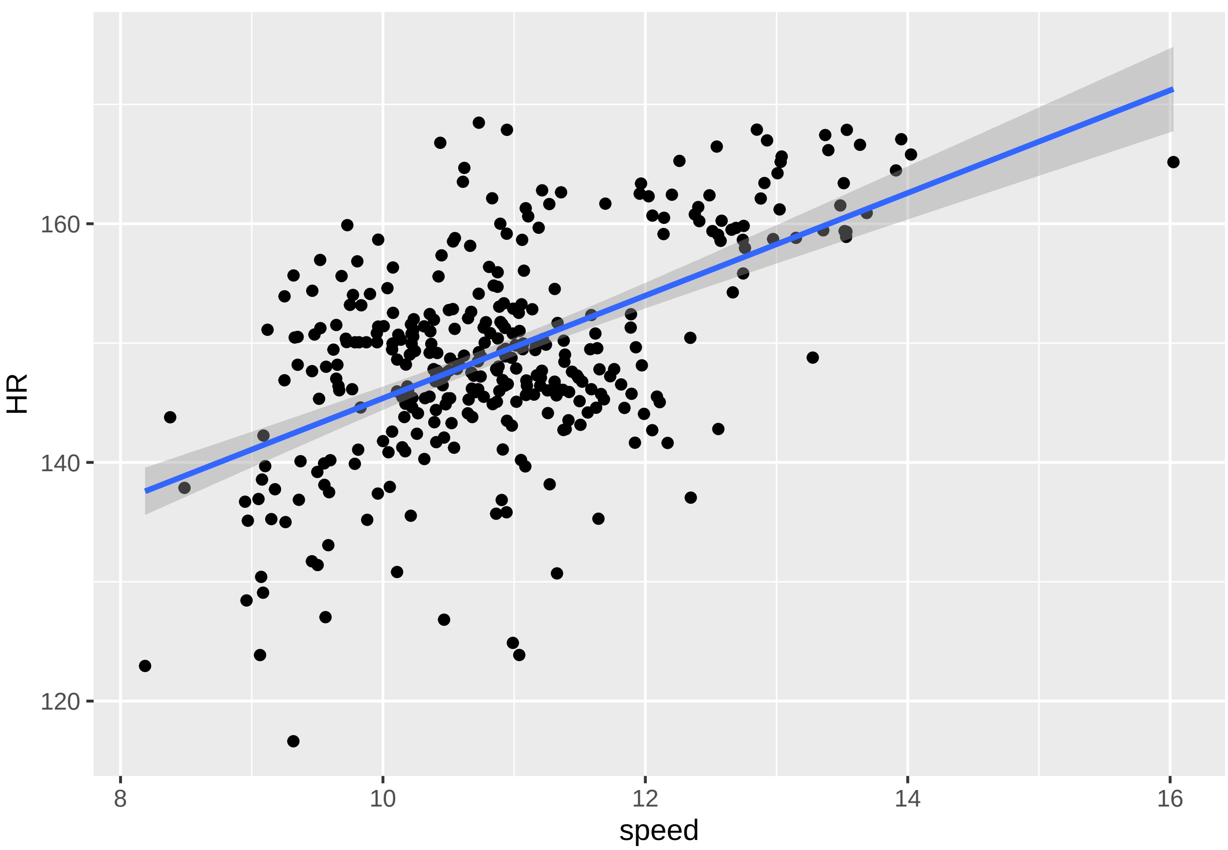
1b

```r
head(data1)
```

```
## # A tibble: 6 x 2
##    pace    HR
##   <dbl> <dbl>
## 1  5.44  124.
## 2  5.03  142.
## 3  5.28  146.
## 4  5.27  143.
## 5  5.00  144.
## 6  5.31  147.
```

```r
data1$speed <- 60/data1$pace
m2 <- ggplot(data1, aes(x = speed, y = HR)) +
  geom_point()+
  geom_smooth(method = "lm")
m2
```

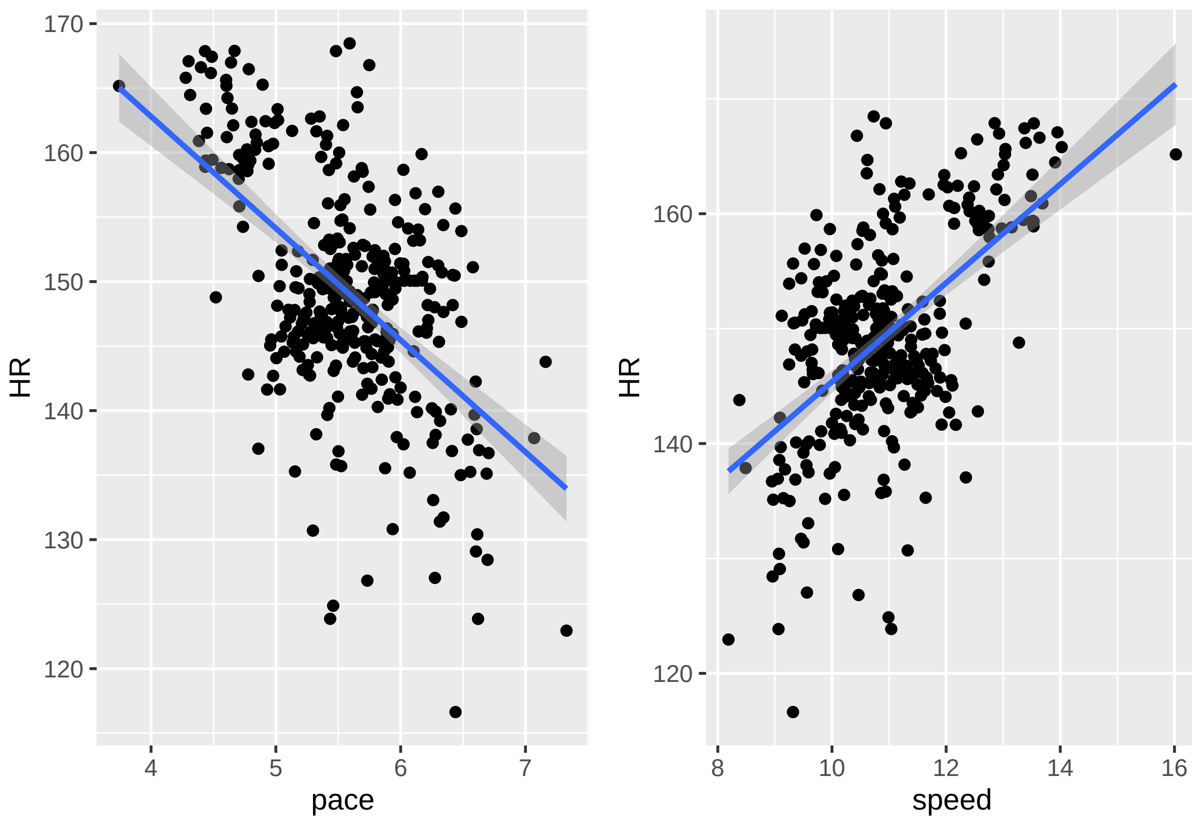
```
## `geom_smooth()` using formula = 'y ~ x'
```



```r
# beide :

m1 | m2
```

```
## `geom_smooth()` using formula = 'y ~ x'
## `geom_smooth()` using formula = 'y ~ x'
```



1d

```r
data1d <- data1 %>% mutate(HRbps = HR * 1/60, speedMi = pace* 1/ 1.61)
head(data1d)
```

```
## # A tibble: 6 x 5
##    pace    HR speed HRbps speedMi
##   <dbl> <dbl> <dbl> <dbl>   <dbl>
## 1  5.44  124.  11.0  2.06    3.38
## 2  5.03  142.  11.9  2.36    3.13
## 3  5.28  146.  11.4  2.43    3.28
## 4  5.27  143.  11.4  2.38    3.28
## 5  5.00  144.  12.0  2.40    3.11
## 6  5.31  147.  11.3  2.45    3.30
```

2

2b

```
set.seed(467)

# ist die n = 10 wichtig?
X <- runif(10000, 0, 10)
Y <- -2 +3.5*X + rnorm(10000, 0, sqrt(10))

variance2 <- 10
n <- 10000 # ANzhal Datenpunkte

simdata <- data.frame(X, Y)
head(simdata)
```

```
##           X          Y
## 1 2.8765086 -0.3991255
## 2 1.7791511  1.2806512
## 3 3.1329369  4.6587175
## 4 0.1047816 -4.1998386
## 5 4.9475554  8.1936600
## 6 9.9074212 32.5172602
```

```
# Varianz Dach (warum ist ds jetzt die wahre? die Werte sind doch geschätzt?)
b1dach <- 10/(n*var(X))
b1dach
```

```
## [1] 0.0001220042
```

```
b0dach <- 10*((1/n)+ ((mean(X))^2)/n*var(X))
b0dach
```

```
## [1] 0.2062437
```

Jetzt zu ermitteln: wie ist beta0 und beta1 verteilt? -> aus Daten lm Modell fitten (10 000 mal wiederholen)

```
reps <- 10000
fit <- matrix(ncol = 2, nrow = reps)
for (i in 1:reps) {
  # wählt zufällig aus unsren Daten 10 Datenpunkte heraus (Zeilen), aber why 10?
  # -> Anzahl an x Werten
  sample_data <- simdata[sample(1:10000, 10),]

  # Aus diesen Daten ein lineares Modell fitten & Koeffizientenj beta extrahieren
  fit[i, ] <- lm(X~Y, data = sample_data)$coefficients
}
head(sample_data)
```

```
##             X          Y
```

```
## 9564 8.365031 28.9303473
## 3668 7.240353 19.5995343
## 7467 9.597639 29.4367143
## 3275 1.187306  0.9759016
## 9418 1.299909  4.0792287
## 969  2.203141  6.1734223
```

```
head(simdata)
```

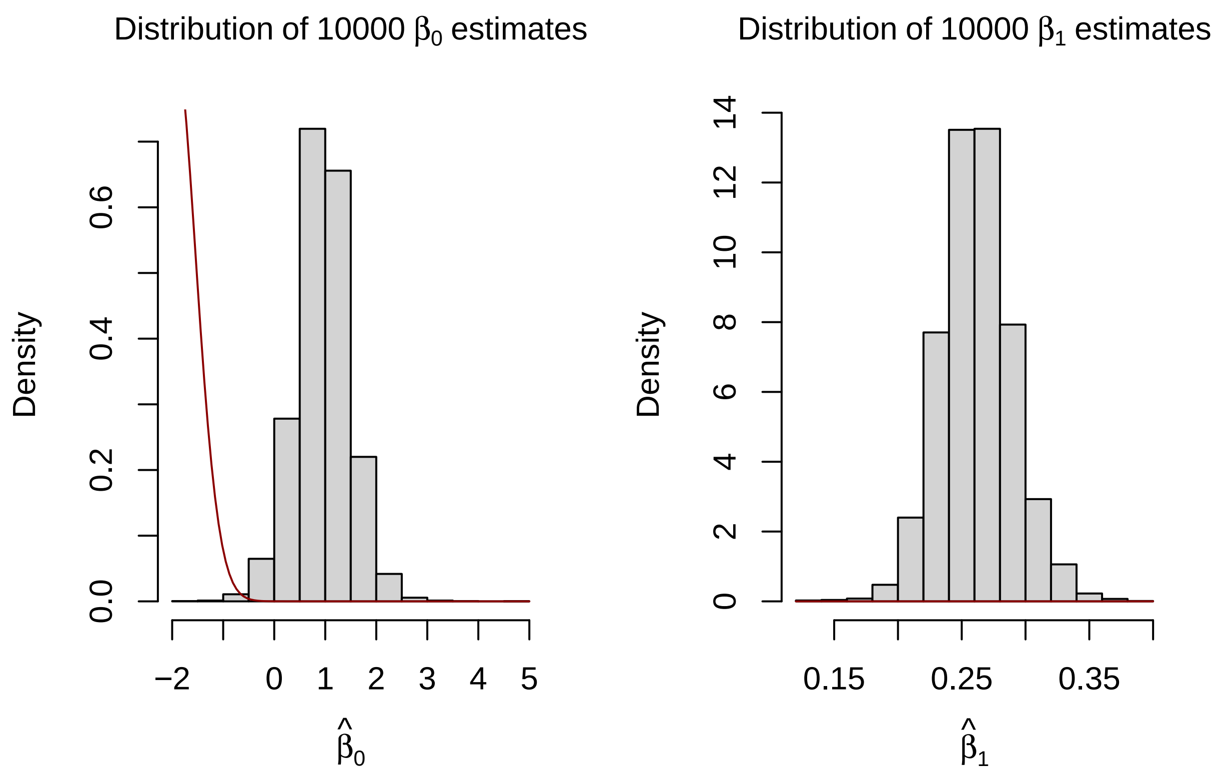
```
##           X          Y
## 1 2.8765086 -0.3991255
## 2 1.7791511  1.2806512
## 3 3.1329369  4.6587175
## 4 0.1047816 -4.1998386
## 5 4.9475554  8.1936600
## 6 9.9074212 32.5172602
```

```
head(fit)
```

```
##           [,1]      [,2]
## [1,] 0.5329718 0.3036812
## [2,] 0.9963975 0.2665542
## [3,] 1.6697623 0.2458257
## [4,] 1.1944555 0.2718532
## [5,] 0.3373230 0.2882058
## [6,] 0.8547328 0.2631829
```

Grafischer Vergleich

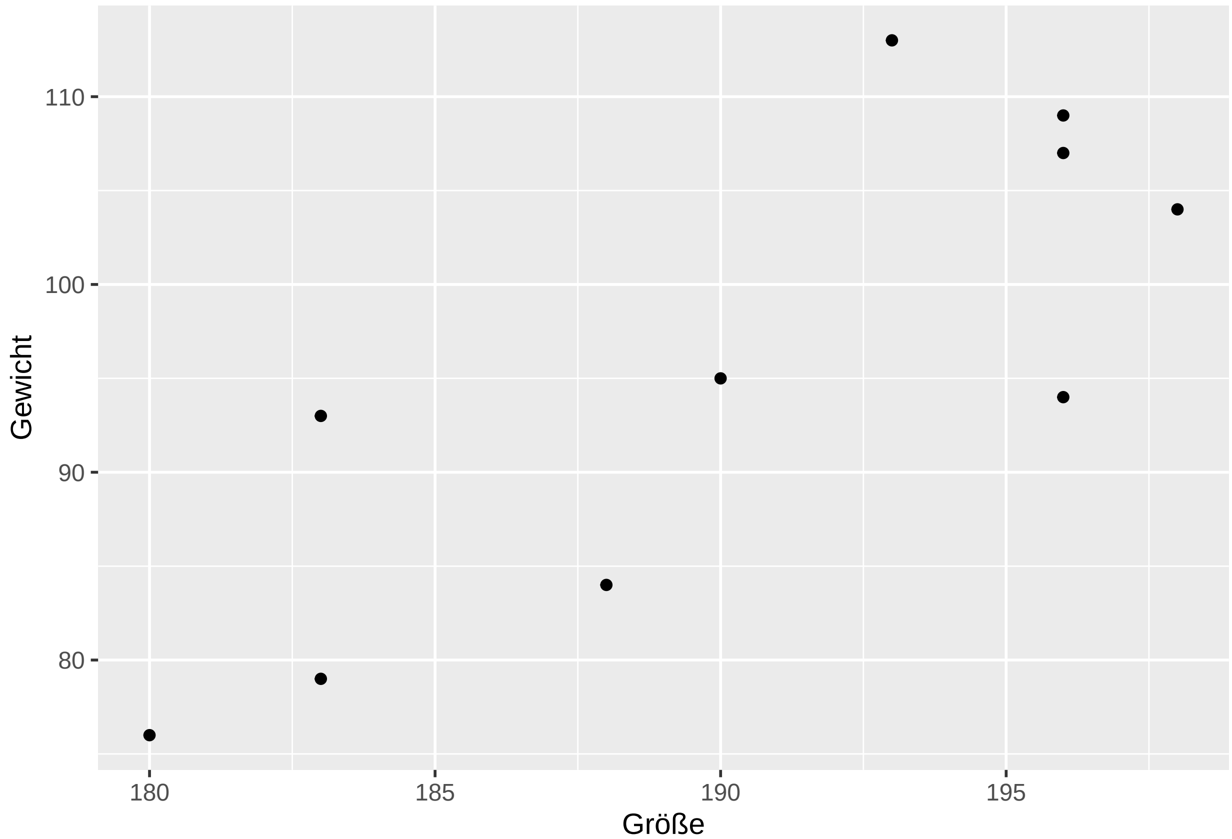
```
par(mfrow = c(1, 2))
# Achsenabschnitt:
hist(x = fit[, 1], cex.main = 1,
     # titel & Achsen
main = bquote(Distribution ~ of ~ 10000 ~ beta[0] ~ estimates),
xlab = bquote(hat(beta)[0]), freq = FALSE)
curve(dnorm(x = x, mean = -2, sd = sqrt(b0dach)), add = TRUE,
col = "darkred")
# Steigungsparameter:
hist(x = fit[, 2], cex.main = 1,
main = bquote(Distribution ~ of ~ 10000 ~ beta[1] ~ estimates),
xlab = bquote(hat(beta)[1]), freq = FALSE)
curve(dnorm(x = x, mean = 3.5, sd = sqrt(b1dach)), add = TRUE,
col = "darkred")
```



3

a

```
data3 <- data.frame("Größe" = c(198, 188, 196, 190, 180, 183, 196, 196, 193, 183),
                    "Gewicht" = c(104, 84, 107, 95, 76, 79, 109, 94, 113, 93))
ggplot(data3, aes(Größe, Gewicht)) +
  geom_point()
```



positiver Zusammenhang erkennbar

b

```
y_strich <- mean(data3$Gewicht)
gewicht <- data3$Gewicht
größe <- data3$Größe
x_strich <- mean(größe)
y_strich
```

```
## [1] 95.4
```

```
sst <- sum((gewicht - y_strich)^2)
sst
```

```
## [1] 1486.4
```

c

```
n <- 10
beta1 <- cov(gewicht, größe)/var(größe)
beta0 <- y_strich - beta1*x_strich
beta0
```

```
## [1] -209.7972
```

```r
beta1
```

```
## [1] 1.603769
```

d

```r
y_dach <- größe*beta1 + beta0

sse <- sum((gewicht - y_dach)^2)

1 - sse/sst
```

```
## [1] 0.6611877
```

e

```r
model3e <- lm(Gewicht ~ Größe, data3)
summary(model3e)
```

```
##
## Call:
## lm(formula = Gewicht ~ Größe, data = data3)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -10.541  -4.457  -1.400   3.958  13.270
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept) -209.7972    77.2826  -2.715  0.02647 *
## Größe          1.6038     0.4059   3.951  0.00423 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 7.934 on 8 degrees of freedom
## Multiple R-squared:  0.6612, Adjusted R-squared:  0.6188
## F-statistic: 15.61 on 1 and 8 DF,  p-value: 0.004229
```