?bung 11

Explorative Datenanalyse und Visualisierung

Wintersemester 2019

S. D?hler (FBMN, h_da)

Name:

Punkte:

Aufgabe 24. Der Datensatz 'mcycle' im R-Paket 'MASS' enth?lt Daten mit den Merkmalen 'accel' und 'times'.

a) Machen Sie sich mit der Herkunft und Bedeutung des Datensatzes vertraut und beschreiben Sie diese mit Ihren eigenen Worten.

b) Plotten Sie die Daten und f?hren Sie eine lineare Regression durch. Diskutieren Sie, ob eine lineare Regression f?r diesen Datensatz angebracht ist.

c) Schreiben Sie eine shiny-Application analog zur Aufgabe 17, die es dem Benutzer erlaubt, unter verscheidenen Kernen und Bandbreiten auszuw?hlen.

Aufgabe 24a)

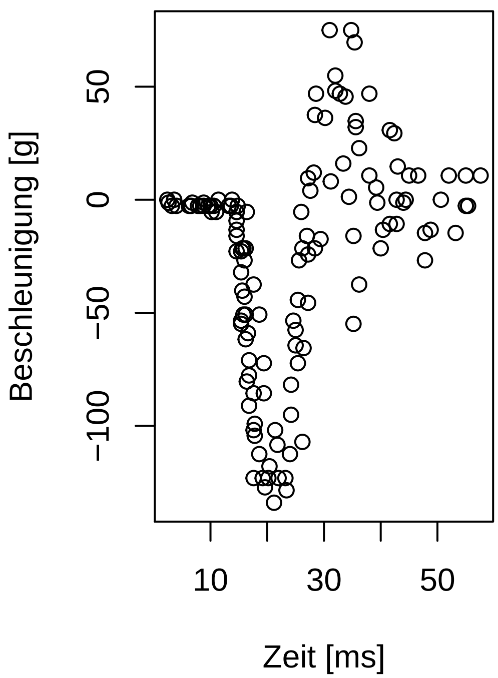
```r
library("MASS")
help(mcycle)
```

Dieser Datensatz simuliert einen Motorradunfall, um das Beschleunigungsverhalten eines Kopfes zu analysieren. Die Simulation wird in zwei Spalten angegeben. Die Spalte "times" beinhaltet die Zeit nach dem Aufprall in Millisekunden und die Spalte "accel" (f?r acceleration / Beschleunigung) ist die Kraft der Geschwindigkeit(Beschleunigung) bzw. G-Kraft, die sich auf dem Kopf des Motorradfahrers zu der Zeit aus Spalte "times" auswirkt.

Aufgabe 24b)

```r
library(lme4)
df = mcycle
plot(df, main = "Beschleunigung des Kopfes nach Aufprall",
     xlab = "Zeit [ms]", ylab = "Beschleunigung [g]")
```

schleunigung des Kopfes nach A




```r
model.1 <- lm(accel ~ times, data = df)
summary(model.1)
#>
#> Call:
#> lm(formula = accel ~ times, data = df)
#>
#> Residuals:
#>      Min       1Q   Median       3Q      Max
#> -104.114  -25.926    4.582   36.163   94.197
#>
#> Coefficients:
#>             Estimate Std. Error t value Pr(>|t|)
#> (Intercept)  -53.008      8.713  -6.084  1.2e-08 ***
#> times          1.091      0.307   3.552 0.000532 ***
#> ---
#> Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
#>
```

```
#> Residual standard error: 46.33 on 131 degrees of freedom
#> Multiple R-squared:  0.08785,    Adjusted R-squared:  0.08089
#> F-statistic: 12.62 on 1 and 131 DF,  p-value: 0.0005318
```

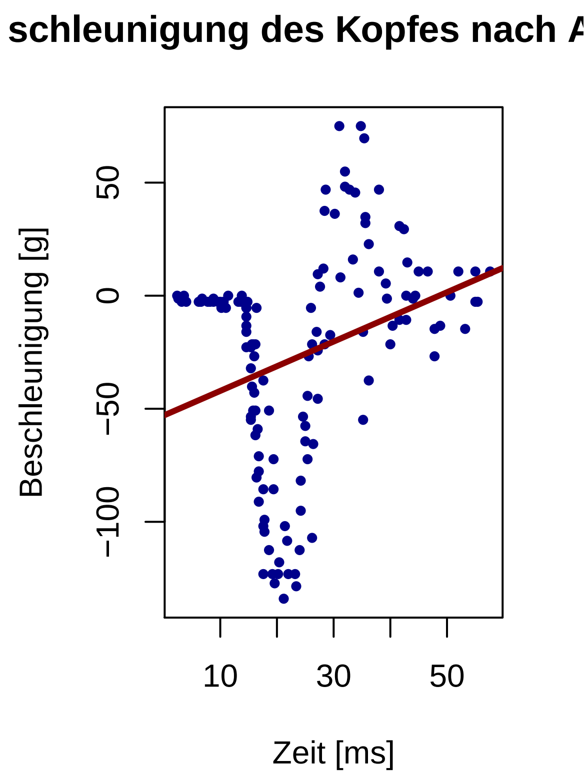
Bereits bei der Darstellung der G-Kraft in Abh?ngigkeit der Zeit, wird im Scatterplot ein nichtlinearer Zusammenhang deutlich. Der Scatterplot zeigt zun?chst, dass die G-Kraft etwa nach 12ms stark abnimmt, bei 20ms ein Minimum von etwa -100g erreicht, anschlie?end stark zunimmt (?ber die Baseline) und bei etwa 30ms einen Peak von (+)50g erreicht, und anschlie?end etwas langsamer auf 0g zur?ckf?llt.

Angewendet auf unseren Motorradunfall bedeutet das, dass der Fahrer ab ca. 15ms in die Luft abhebt und dementsprechend keine Kraft auf diesen wirkt(gegen die Erdanziehung) und nach ca. 10 ms zu Boden f?llt, was f?r einen Anstieg der G-Kraft spricht. Bei ca. 35ms kollidiert er mit den Boden und hast somit maximale G-Kraft.Danach sinkt die G-Kraft wieder. Einen linearen Zusammenhang ist somit ausgeschlossen.

Wir f?hren nun eine Lineare Regression (Abh?ngige Variable: Beschleunigung, Unabh?ngige Variable: Zeit) durch. Wir sehen zun?chst, dass das Regressionsmodell signifikant wird (p=0.0005). Der Intercept wurde auf -53g gesch?tzt (signifikant mit p<0.001), Die Steigung auf etwa 1.1g/ms (signifikant mit p<0.001).

Wenn wir uns die Regressionsgerade in unseren Scatterplot einzeichnen, …

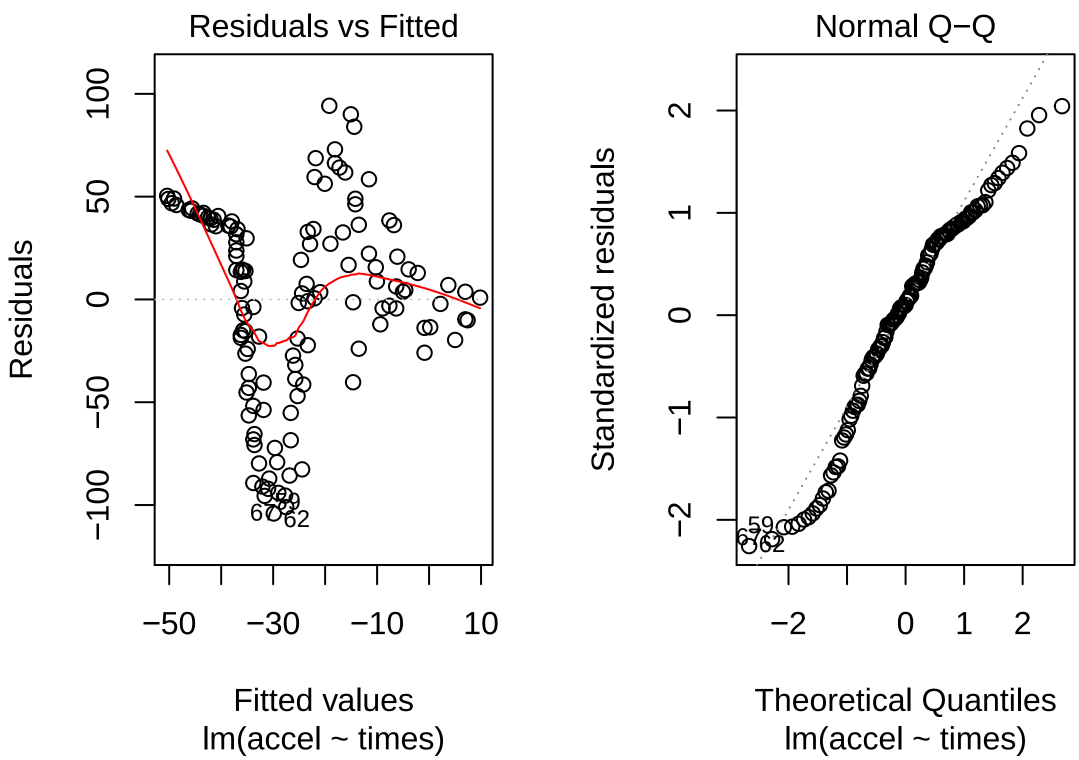
```r
plot(df, main = "Beschleunigung des Kopfes nach Aufprall",
     xlab = "Zeit [ms]", ylab = "Beschleunigung [g]",
     pch = 16, cex = 0.7,  col = "darkblue")
abline(model.1, col = "darkred", lwd = 3)
```

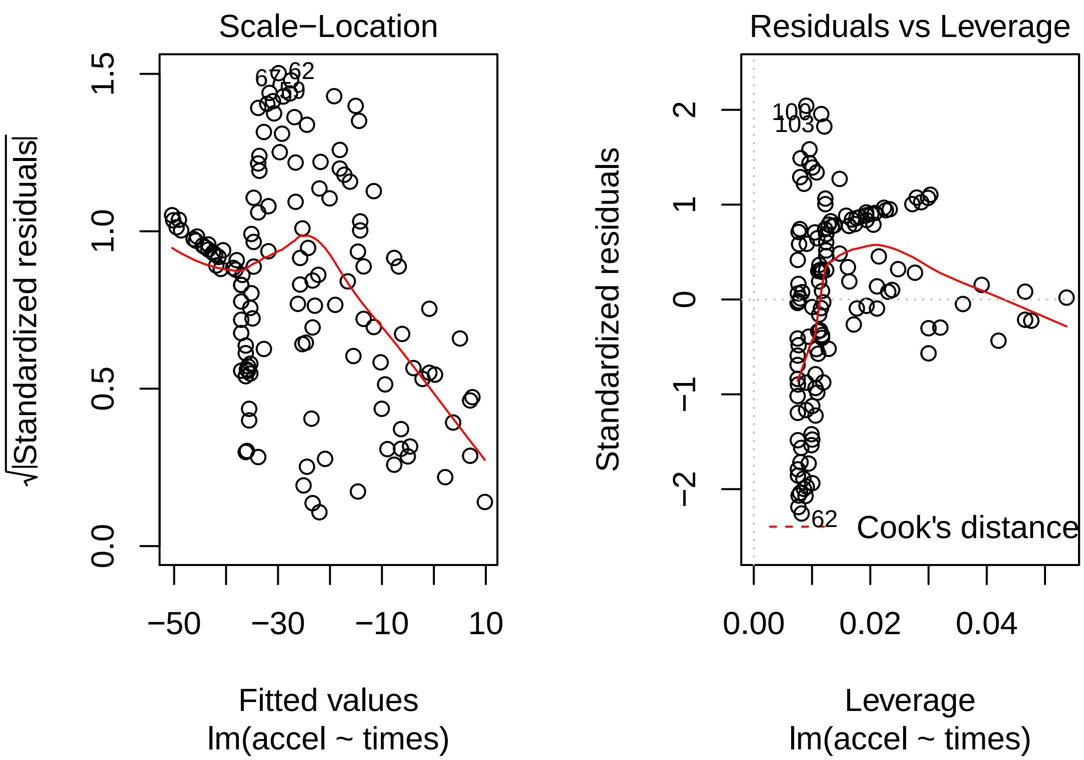


. . . sehen wir, dass die Gerade das Verhalten der Datenpunkte nicht sehr gut erfasst. Wir erkennen klar, dass hier ein nicht-linearer Zusammenhang besteht. Die meisten Punkte sind meist weit entfernt von der Regressionsgerade und kann das Muster nicht nachbilden.

Wenn wir uns aber das R-Squared unseres Modells anschauen, f?llt uns schon auf, dass das Modell nicht sehr gut ist um unsere Daten zu beschreiben, denn R-Squared ist <0.1. Das bedeutet unser Modell beschreibt gerade mal weniger als 10% der Variabilit?t in den Daten.

```r
plot(model.1)
```


Residuals vs Fitted
Residuals
Fitted values
lm(accel ~ times)
−50
−30
−10
10
100
50
0
−50
−100
67
62
Normal Q−Q
Standardized residuals
Theoretical Quantiles
lm(accel ~ times)
−2
0
1
2
−1
59
67
62



Auch die Residualanalyse best?tigt unsere Behauptung.Auch die Residualanalyse best?tigt unsere Vermutung. In der Darstellung der Residuen ist ein Muster zu erkennen, der best?tigt, dass ein nichtlinearer Zusammenhang vorliegt. Ein Nichtlineares Modell (vielleicht ein Polynom h?herer Ordnung) k?nnte hier besser passen.

Aufgabe 24c) Shiny Application

```
library(shiny)
# Define UI for application that draws a histogram
ui <- fluidPage(

    titlePanel(" Beschleunigung des Kopfes nach dem Aufprall"),

    #Input
    selectInput(inputId = "Kern",
                label = "W?hle einen Kernsch?tzer aus:",
                choices = c("symmetric","gaussian"),
                selected = "symmetric"),
```

```
    sliderInput(inputId = "Bandbreite",
                label = "W?hle eine Bandbreite aus:",
                min=0.01,
                max = 2.00,
                step = 0.01,
                value = 1),

    plotOutput(outputId = "main_plot", height = "300px"),
    )


# Define server logic required to draw a histogram
server<-function(input, output) {

    output$main_plot <- renderPlot({

        plot(x=df$times,y=df$accel,ylab = "G-Kraft in Abh?ngigkeit der Zeit", xlab = "Zeit i
      lines(loess.smooth(x = df$times, y= df$accel, span = input$Bandbreite, family = input$
    })
}
# Run the application
shinyApp(ui = ui, server = server)
```

Shiny applications not supported in static R Markdown documents

Aufgabe 25. Sie sollen die Zeitreihen 'globtemp' und 'gtemp' aus dem Paket 'astsa' (explorativ) analysieren.

a) Beschreiben Sie kurz in Ihren eigenen Worten die Bedeutung, Herkunft und Erhebung der Daten.

b) F?hren Sie zun?chst eine lineare Regression durch und interpretieren Sie die entsprechende ANOVA-Tabelle. Erzeugen Sie ein gemeinsames Grafik-Panel, das aus 2 Grafiken besteht:

   – die Zeitreihe sowie die lineare Regression in einem gemeinsamen Plot

   – ein QQ-Plot der Residuen. Dikutieren Sie, ob die Residuen normalverteilt sind.

   Diskutieren Sie ggf. Unterschiede, die sich bei 'globtemp' gegen?ber 'gtemp' ergeben.

c) Bearbeiten Sie Aufgabe b), indem Sie jedoch als Regressionssch?tzer Kern-Sch?tzer verwenden. Probieren Sie verschiedene Bandbreiten aus.

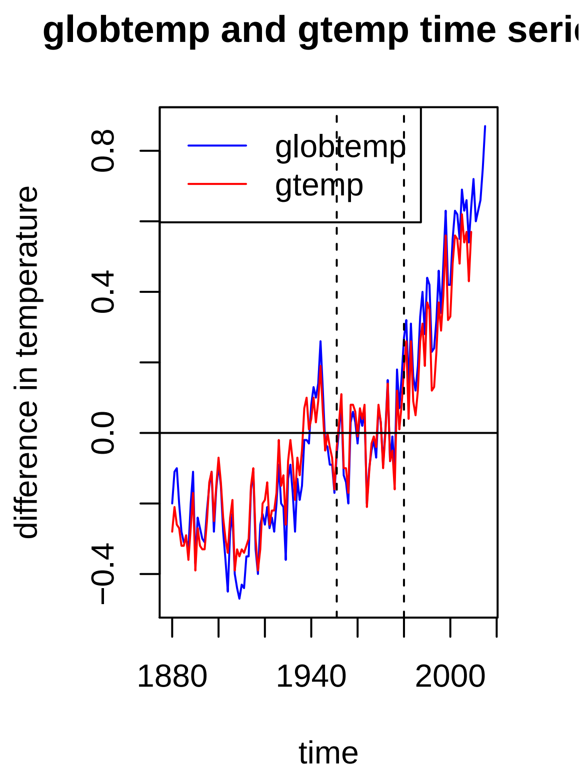
d) Probieren Sie verschiedene symmetrische (und gewichtete) Filter aus, die Sie auf die Zeitreihe anwenden.

e) Schrieben Sie shiny-Apps, die Aufgaben c) und d) implementieren.

f) Wenden Sie die Holt-Winters-Methode auf die zeitreihe an, um z.B. eine exponentielle Gl?ttung der Zeitreihe zu erzielen.

Aufgabe 25a)

```
library("astsa")
df1 = globtemp
df2 = gtemp
library("astsa")
?globtemp
?gtemp
{
plot(globtemp, col = "blue", main = "globtemp and gtemp time series", ylab = "difference in
lines(gtemp, col = "red")
legend("topleft", c("globtemp","gtemp"), col = c("blue","red"), lty = 1)
abline(v = 1951, lty = 14)
abline(v = 1980, lty = 14)
abline(h = 0)
}
```



globtemp:

Die Daten reichen von 1880 bis 2015 und beinhalten Mittelwerte der globalen Temperaturschwankungen zwischen Land und Ozean (in Einheit Grad Celsius). Als Baseline wurde die mittleren Temperaturunterschiede von 1951-1980 verwendet (im Plot mit vertikalten, gestrichelten Linien eingezeichnet). Alle Temperaturunterschiede beziehen sich auf die genormten Temperaturunterschiede in diesem Zeitraum.

gtemp: Dies ist eine ?ltere Version des Datensatzes. Er wird nur noch als Referenz verwendet. Der Globtemp Datensatz beinhaltet neue Zahlen (+ ein paar Jahre mehr), einige Daten wurden wegen mangelnder Qualität entfernt.

Man erkennt schon leichte Unterschiede der beiden Zeitreihen im Plot (haupts?chlich au?erhalb des Jahresbereiches, auf welchen normiert wurde).

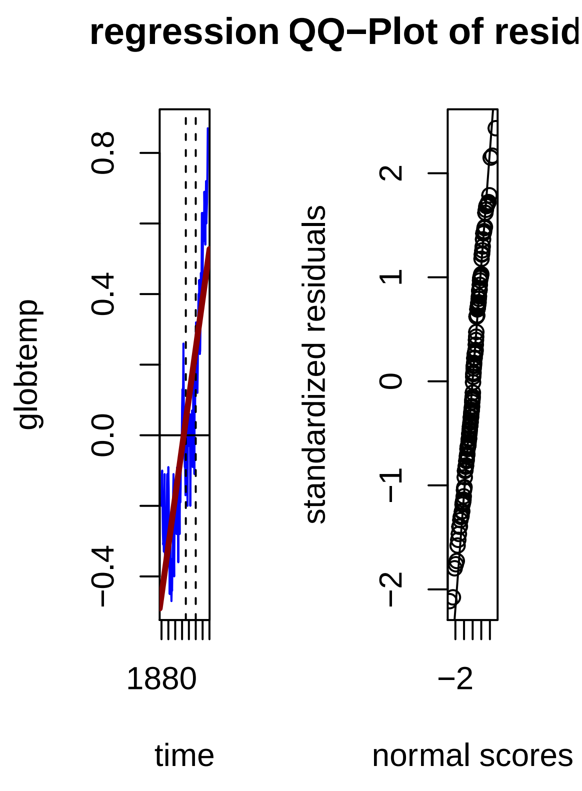
Aufgabe 25b)

Globtemp

```r
df1$time = as.numeric(time(globtemp))
#> Warning in df1$time = as.numeric(time(globtemp)): Wandle linke Seite in
#> eine Liste um
df1$temperature = as.numeric(globtemp)

df2$time = as.numeric(time(gtemp))
#> Warning in df2$time = as.numeric(time(gtemp)): Wandle linke Seite in eine
#> Liste um
df2$temperature = as.numeric(gtemp)

model.25.b.1 = lm(as.numeric(globtemp) ~ as.numeric(time(globtemp)))
summary(model.25.b.1)
#>
#> Call:
#> lm(formula = as.numeric(globtemp) ~ as.numeric(time(globtemp)))
#>
#> Residuals:
#>      Min       1Q   Median       3Q      Max
#> -0.33363 -0.11470 -0.02466  0.11932  0.38017
#>
#> Coefficients:
#>                              Estimate Std. Error t value Pr(>|t|)
#> (Intercept)                -1.358e+01  6.747e-01  -20.13   <2e-16 ***
#> as.numeric(time(globtemp))  6.984e-03  3.464e-04   20.16   <2e-16 ***
#> ---
#> Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
#>
#> Residual standard error: 0.1586 on 134 degrees of freedom
#> Multiple R-squared:  0.7521, Adjusted R-squared:  0.7503
#> F-statistic: 406.6 on 1 and 134 DF,  p-value: < 2.2e-16
anova(model.25.b.1)
#> Analysis of Variance Table
#>
#> Response: as.numeric(globtemp)
#>                             Df  Sum Sq Mean Sq F value    Pr(>F)
#> as.numeric(time(globtemp))   1 10.2253 10.2253  406.62 < 2.2e-16 ***
#> Residuals                  134  3.3697  0.0251
#> ---
#> Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

Im globtemp Datensatz: Wir beschreiben sch?tzen die Temperaturunterschiede (DV) aus dem Jahr (IV) und erhalten einen signifikanten linearen Zusammenhang mit F(1,134)=406, und p=2.2*10^-16. Die mittleren Quadratsummen der Temperaturdifferenzen sind um einiges h?her als die Quadratsummen der Residuen.

```
par(mfrow=(c(1,2)))
res = rstandard(model.25.b.1)
{
plot(globtemp, col = "blue", main = "regression", xlab = "time")
abline(v = 1951, lty = 14)
abline(v = 1980, lty = 14)
abline(h = 0)
abline(model.25.b.1, col = "darkred", lwd = 3)
}
{
qqnorm(res,
   ylab="standardized residuals",
   xlab="normal scores",
   main="QQ-Plot of residuals")
qqline(res)
}
```



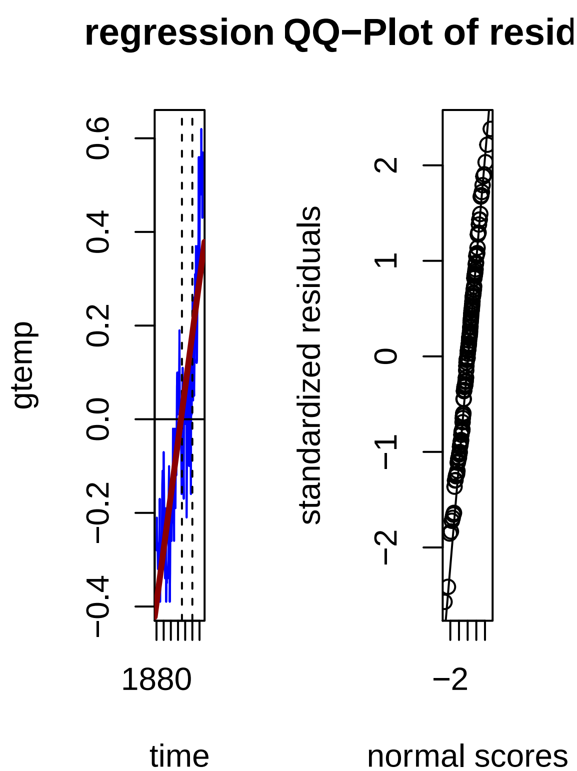
In der linken Abbildungen sehen wir die Zeitreihe mit der Regressionsgeraden.

Die Regressionsgerade erfasst zwar den groben Trend, trotz allem erfasst sie nicht alle Zeitr?ume sehr genau. V.a. der Zeitraum, auf welchen die Daten normiert wurden, ist von der Gerade gar nicht gut erfasst. In der rechten den QQ-Plot der standardisierten Residuen. Die Residuen verteilen sich grob an der Winkelhalbierenden, was f?r eine Normalverteilung spricht. In den Extremen gibt es jedoch Abweichungen von der Winkelhalbierenden / Normalverteilung. Wir denken, es liegt keine optimale Normalverteilung der Residuen vor.

gtemp

```
model.25.b.2 = lm(as.numeric(gtemp) ~ as.numeric(time(gtemp)))
summary(model.25.b.2)
#>
#> Call:
#> lm(formula = as.numeric(gtemp) ~ as.numeric(time(gtemp)))
#>
#> Residuals:
#>      Min       1Q   Median       3Q      Max
#> -0.31946 -0.09722  0.00084  0.08245  0.29383
#>
#> Coefficients:
#>                           Estimate Std. Error t value Pr(>|t|)
#> (Intercept)             -1.120e+01  5.689e-01  -19.69   <2e-16 ***
#> as.numeric(time(gtemp))  5.749e-03  2.925e-04   19.65   <2e-16 ***
#> ---
#> Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
#>
#> Residual standard error: 0.1251 on 128 degrees of freedom
#> Multiple R-squared:  0.7511, Adjusted R-squared:  0.7492
#> F-statistic: 386.3 on 1 and 128 DF,  p-value: < 2.2e-16
anova(model.25.b.2)
#> Analysis of Variance Table
#>
#> Response: as.numeric(gtemp)
#>                          Df Sum Sq Mean Sq F value    Pr(>F)
#> as.numeric(time(gtemp))   1 6.0496  6.0496  386.25 < 2.2e-16 ***
#> Residuals               128 2.0048  0.0157
#> ---
#> Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

par(mfrow=(c(1,2)))
res = rstandard(model.25.b.2)
{
plot(gtemp, col = "blue", main = "regression", xlab = "time")
abline(v = 1951, lty = 14)
abline(v = 1980, lty = 14)
```

```r
abline(h = 0)
abline(model.25.b.2, col = "darkred", lwd = 3)
}
{
qqnorm(res,
   ylab="standardized residuals",
   xlab="normal scores",
   main="QQ-Plot of residuals")
qqline(res)
}
```



Wenn wir den (alten) Datensatz "gtemp" verwenden, sehen wir ein sehr ?hnliches Bild. Alle Aussagen ?ber die Normalverteilung und ?ber den Fit der Regressionsgeraden treffen auch hier zu. in den unteren Extremen ist die Abweichung von der Regressionsgerade jedoch leicht schwächer als im Globtemp Datensatz. Der F-Wert ist fast so hoch wie im "neuen" Datensatz, und der lineare Zusammenhang zwischen Temperaturunterschied und Zeit ist ebenfalls hoch signifikant. Die Steigung ist ein wenig geringer ausgefallen, als bei dem "globtemp" Datensatz

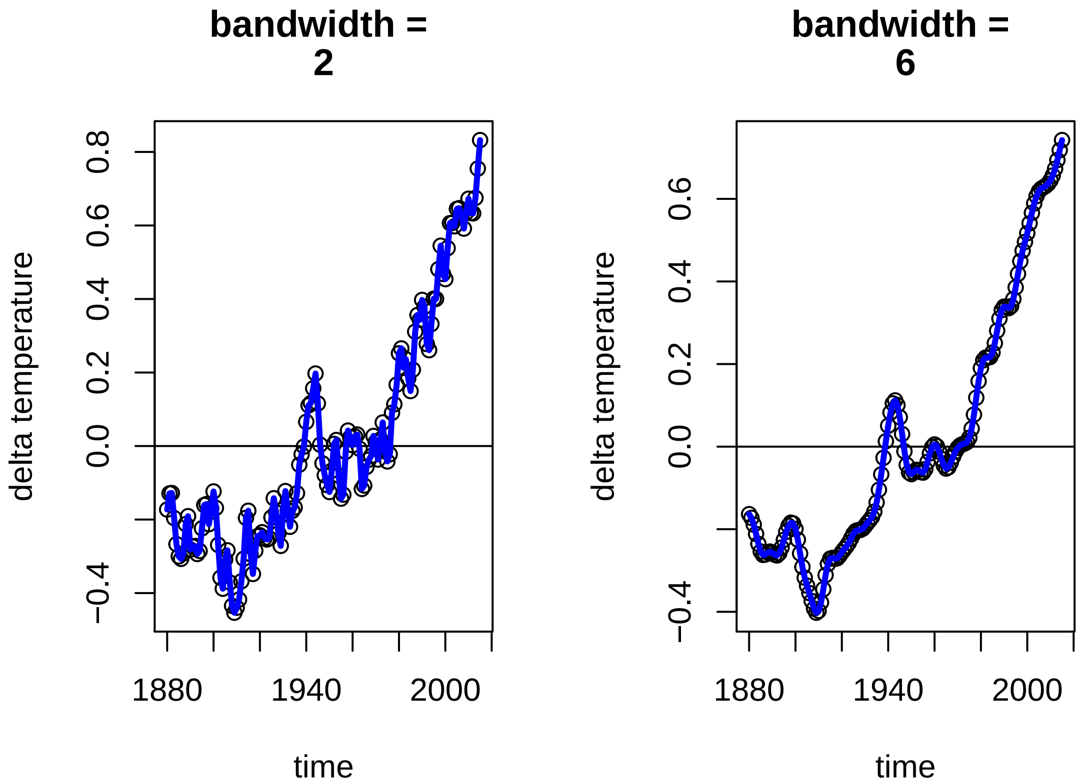
(0.0057 gegen?ber 0.0069). Die Interpretation der Steigung w?re, dass mit jedem Jahr der Temperaturunterschied Meer/Land um 0.0057 (Globtemp) bzw. 0.0069 (Gtemp) ?C zunimmt im Vergleich zum Vorjahr. Wir erkennen auch in beiden Datens?tzen, dass die Steigung (nicht nur das Modell als ganzes) sehr signifikant ist. Wir k?nnen hier interpretieren, dass ein signifikanter, positiver, linearer Zusammenhang vorhanden ist, bzw. der Temperaturunterschied ?ber die Zeit signifikant gr??er wurde.

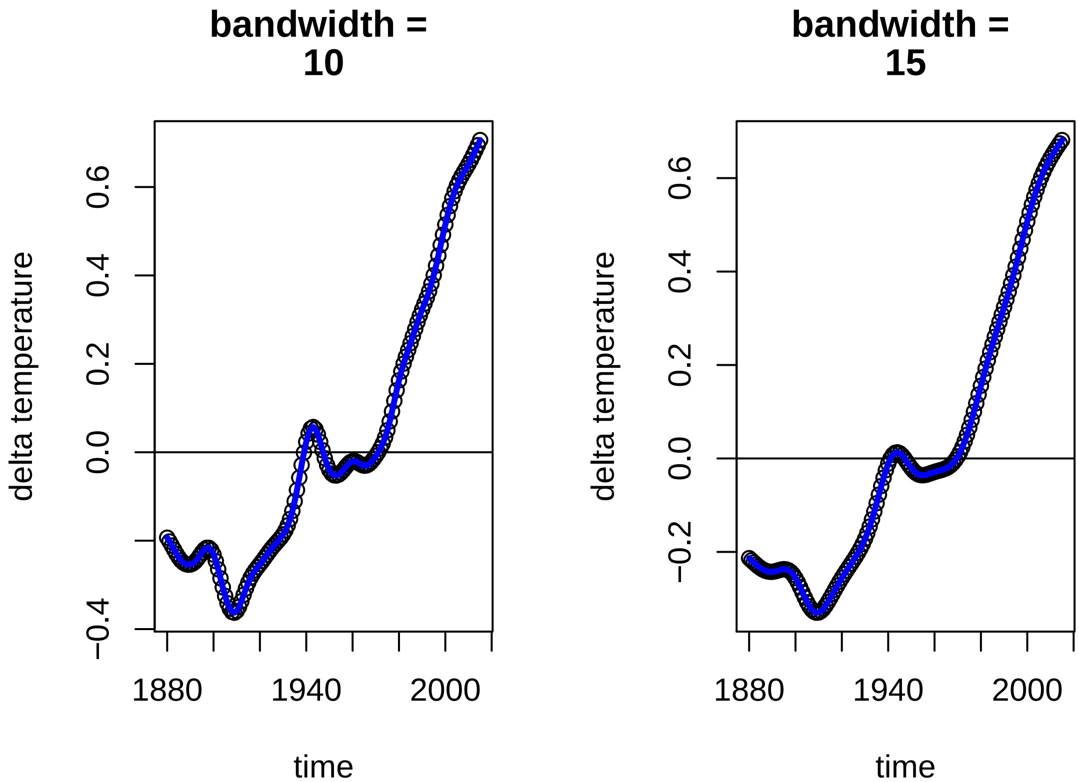
Aufgabe c)

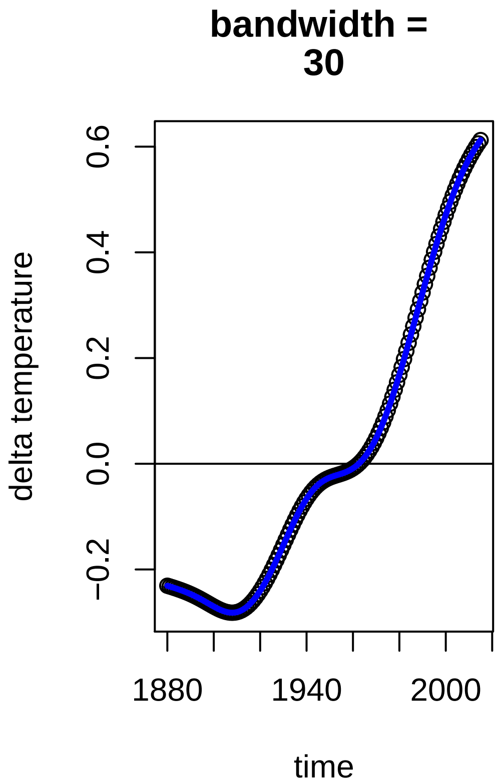
Aufgrund der ?hnlichkeit der beiden Datens?tze rechnen wir im folgenden nur noch mit dem "globtemp" Datensatz.

Nataraya-Watson Methode:

```
bwd <- c(2,6,10,15,30)
for (b in bwd){
  ynw = ksmooth(x = as.numeric(time(globtemp)),
                y = as.numeric(globtemp),
                kernel = "normal",
                bandwidth = b
                )

  {
  plot(x = ynw$x, y = ynw$y, main = paste(c("bandwidth = ", b)), xlab = "time", ylab = "delt
  abline(h = 0)
  lines(ynw, col = "blue", lwd = 3)
  }
}
```

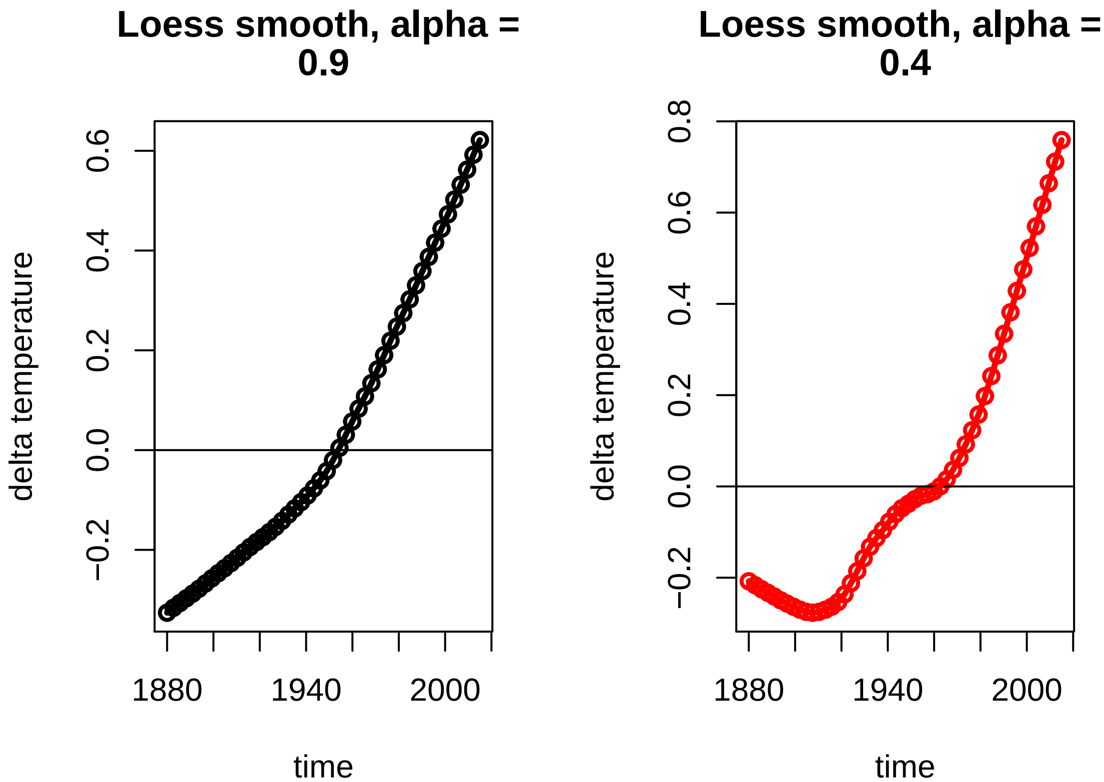

bandwidth =
2
delta temperature
−0.4
0.0
0.2
0.4
0.6
0.8
1880
1940
2000
time
bandwidth =
6
delta temperature
−0.4
0.0
0.2
0.4
0.6
1880
1940
2000
time


bandwidth = 10
delta temperature
0.6
0.4
0.2
0.0
−0.4
1880
1940
2000
time
bandwidth = 15
delta temperature
0.6
0.4
0.2
0.0
−0.2
1880
1940
2000
time

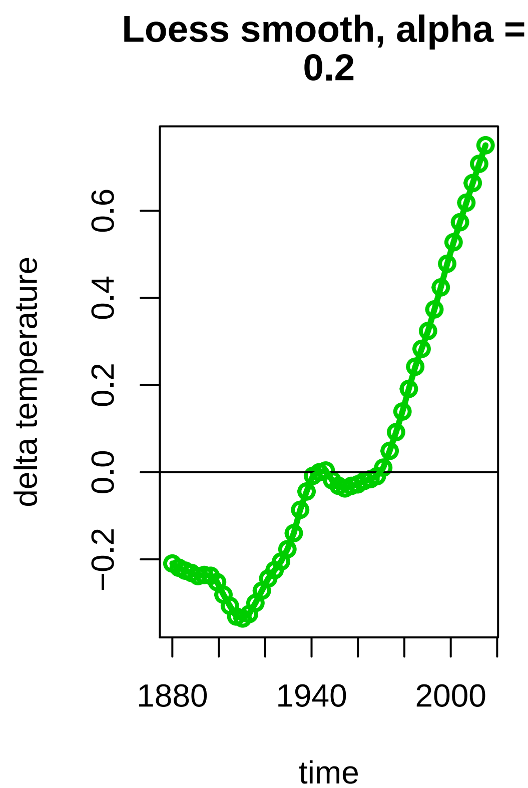


Loess Methode:

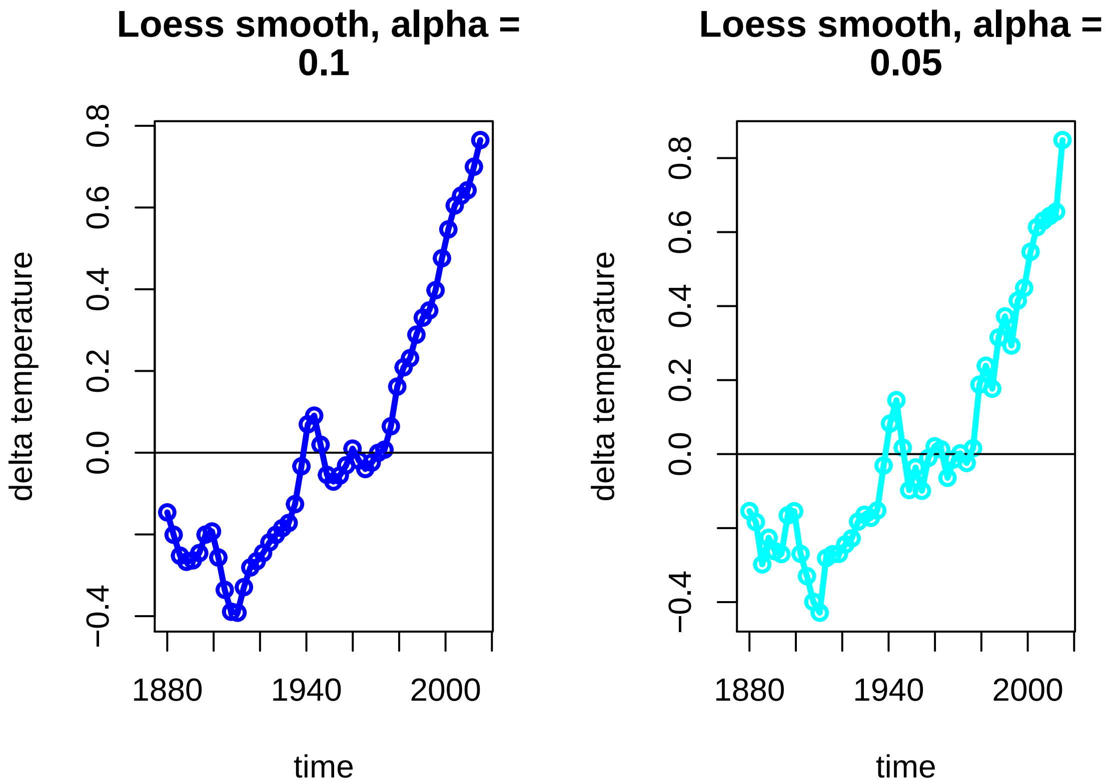
```
alphas <- c(0.9,0.4,0.2,0.1,0.05)
counter <- 0
for (a in alphas){
  counter <- counter + 1
  loessdata <- loess.smooth(y = as.numeric(globtemp), x = as.numeric(time(globtemp)) , model
  plot(x = loessdata$x, y = loessdata$y,
     main = paste(c("Loess smooth, alpha = ",a)),
     xlab = "time",
     ylab = "delta temperature",
     col = counter, #"darkgreen",
     lwd = 2,
     )
  abline(h = 0)
  lines(loessdata, col = counter, lwd = 3)
}
```



Loess smooth, alpha = 0.9
delta temperature
0.6
0.4
0.2
0.0
−0.2
1880
1940
2000
time
Loess smooth, alpha = 0.4
delta temperature
0.8
0.6
0.4
0.2
0.0
−0.2
1880
1940
2000
time

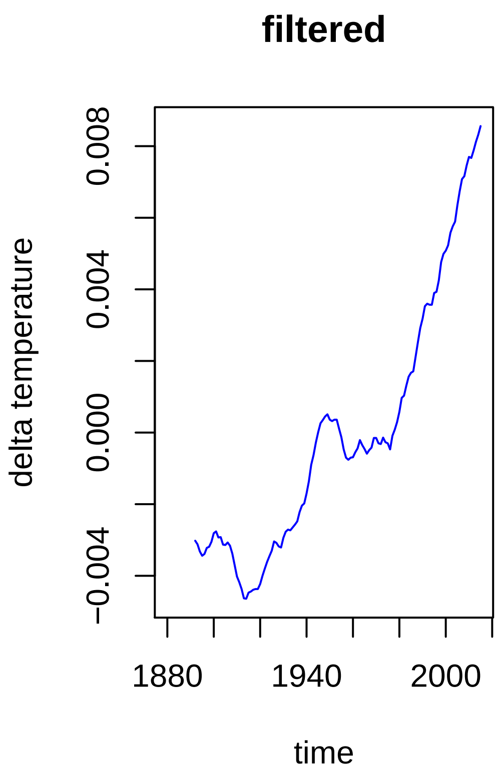



```
#> Warning in simpleLoess(y, x, w, span, degree = degree, parametric =
#> FALSE, : k-d tree limited by memory. ncmax= 200

#> Warning in simpleLoess(y, x, w, span, degree = degree, parametric =
#> FALSE, : k-d tree limited by memory. ncmax= 200

#> Warning in simpleLoess(y, x, w, span, degree = degree, parametric =
#> FALSE, : k-d tree limited by memory. ncmax= 200

#> Warning in simpleLoess(y, x, w, span, degree = degree, parametric =
#> FALSE, : k-d tree limited by memory. ncmax= 200

#> Warning in simpleLoess(y, x, w, span, degree = degree, parametric =
#> FALSE, : k-d tree limited by memory. ncmax= 200
```



Aufgabe 25d)

```
plot(filter(globtemp, filter = rep(0.001,length(globtemp)/10), method = "convolution", sides
     ylab = "delta temperature")
```



Aufgabe 25e)

```
library(shiny)
ui <- fluidPage(

  # Application title
  titlePanel("Globale Temperatur-Schwankung"),

  # Sidebar with a slider input for number of bins
  sidebarLayout(
    sidebarPanel(
      sliderInput("span",
                  "Spannweite Regression:",
                  min = 0,
                  max = 1,
                  value = 0.5),
      textInput("filter", "Filter;", value = "0.1, 0.1, 0.1")
    ),
```

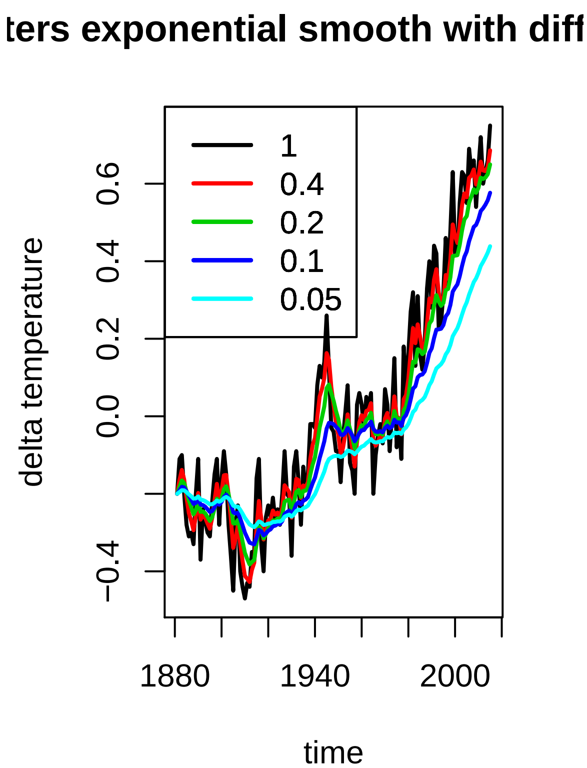
```
    # Show a plot of the generated distribution
    mainPanel(
      plotOutput("distPlot")
    )
  )
)

# Define server logic required to draw a histogram
server <- function(input, output) {

  {

    output$distPlot <- renderPlot({
      inputfilter <- paste(c("c(", input$filter, ")"), collapse = "")
      fil <- eval(parse(text = inputfilter))
      {plot(globtemp, main = "Temperaturschwankungen 1880 - 2015", xlab = "Jahre", ylab = "T
        lines(loess.smooth(x = time(globtemp), y = globtemp, span = input$span), col= "red")
        lines(filter(globtemp, filter = fil, method = "convolution", sides = 2), col = "blue
    })

  }
}

# Run the application
shinyApp(ui = ui, server = server)
```

Shiny applications not supported in static R Markdown documents

Aufgabe 25f)

```
alphas <- c(1.0,0.4,0.2,0.1,0.05)
counter <- 0
for (a in alphas){
  counter <- counter + 1
  hw <- HoltWinters(globtemp, alpha = a, beta = FALSE, gamma = FALSE)
  if (counter == 1){
    plot(fitted(hw)[,1],
       main = "HoltWinters exponential smooth with different alphas",
       xlab = "time",
       ylab = "delta temperature",
       col = counter, #"darkgreen",
       lwd = 2,
       )
  } else {
      lines(fitted(hw)[,1],
```

```
        main = paste(c("alpha = ",a)),
        xlab = "time",
        ylab = "delta temperature",
        col = counter, #"darkgreen",
        lwd = 2,
        )
  }
legend("topleft", legend = alphas, col = seq(length(alphas)), lty = 1, lwd = 2)
}
```



Wir sehen, wie die HoldWinters Methode f?r verschiedene alpha-Werte die Zeitreihe gl?ttet. Bei Alphas von unter 0.1 scheint schon zu "zu viel" aus den Daten weg-gegl?ttet zu sein. Alphas von 1 ver?ndern die Daten nicht.

Anmerkungen/Korrektur
