1 Binäre Regression (I)

Aufgabe 1

Der Datensatz `shuttle` beschreibt für die 23 Space Shuttle Flüge vor dem Challenger-Unglück 1986 die Temperatur ($°F$) zur Startzeit sowie das Auftreten bzw. Nicht-Auftreten einer thermischen Überbeanspruchung eines bestimmten Bauteils. Er enthält folgende Variablen:

`flight`	Nummer des Fluges
`temp`	Temperatur in $°F$
`td`	Thermische Überbeanspruchung (1 = Ja / 0 = Nein)

Laden Sie den Datensatz `shuttle` von der Vorlesungshomepage herunter. Öffnen Sie R, und lesen Sie den Datensatz mit dem Befehl `read.table()` ein. Erzeugen Sie eine zusätzliche Spalte `tempC`, welche die Temperatur in Grad Celsius angibt. Dabei gilt die Umrechnung $T_F = 1.8 \cdot T_C + 32$.

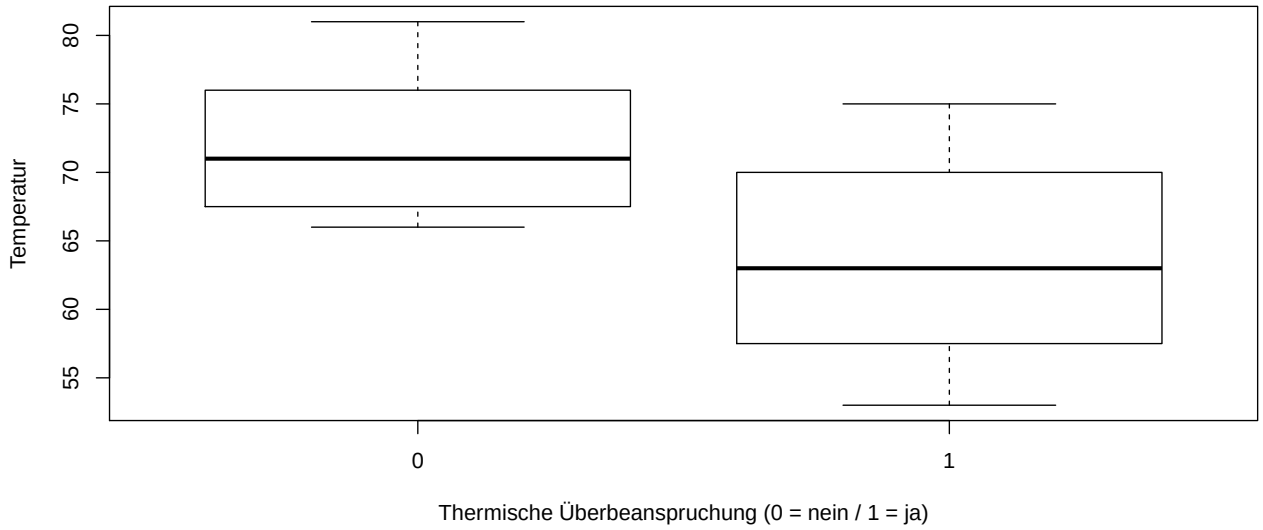
```
## zusatzpakete
library(reshape2)
library(gridExtra)
library(ggplot2)
theme_set(theme_bw())
```

```
# datensatz von der homepage laden
shuttle <- read.table("http://www.statistik.lmu.de/institut/lehrstuhl/semsto/Lehre/GRMWS1415/shuttle.asc"
        header = TRUE)
# temperatur in Celsius hinzufügen
shuttle$tempC <- (shuttle$temp - 32)/1.8
shuttle
```

```
##    flight temp td    tempC
## 1       1   66  0 18.88889
## 2       2   70  1 21.11111
## 3       3   69  0 20.55556
## 4       4   68  0 20.00000
## 5       5   67  0 19.44444
## 6       6   72  0 22.22222
## 7       7   73  0 22.77778
## 8       8   70  0 21.11111
## 9       9   57  1 13.88889
## 10     10   63  1 17.22222
## 11     11   70  1 21.11111
## 12     12   78  0 25.55556
## 13     13   67  0 19.44444
## 14     14   53  1 11.66667
## 15     15   67  0 19.44444
## 16     16   75  0 23.88889
## 17     17   70  0 21.11111
## 18     18   81  0 27.22222
## 19     19   76  0 24.44444
## 20     20   79  0 26.11111
## 21     21   75  1 23.88889
## 22     22   76  0 24.44444
## 23     23   58  1 14.44444
```

(a) **Vergleichen Sie die Temperaturen, die bei `td` = 1 gemessen wurden mit jenen bei `td` = 0. Was ist an einer derartigen Analyse problematisch?**

```
# a)
# summary, boxplot getrennt nach td (0/1)
summary(shuttle$temp[shuttle$td==0])

##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##   66.00   67.75   71.00   72.12   76.00   81.00

summary(shuttle$temp[shuttle$td==1])

##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##   53.00   57.50   63.00   63.71   70.00   75.00

boxplot(temp ~ td, data = shuttle, xlab="Thermische Überbeanspruchung (0 = nein / 1 = ja)",
ylab="Temperatur")
```



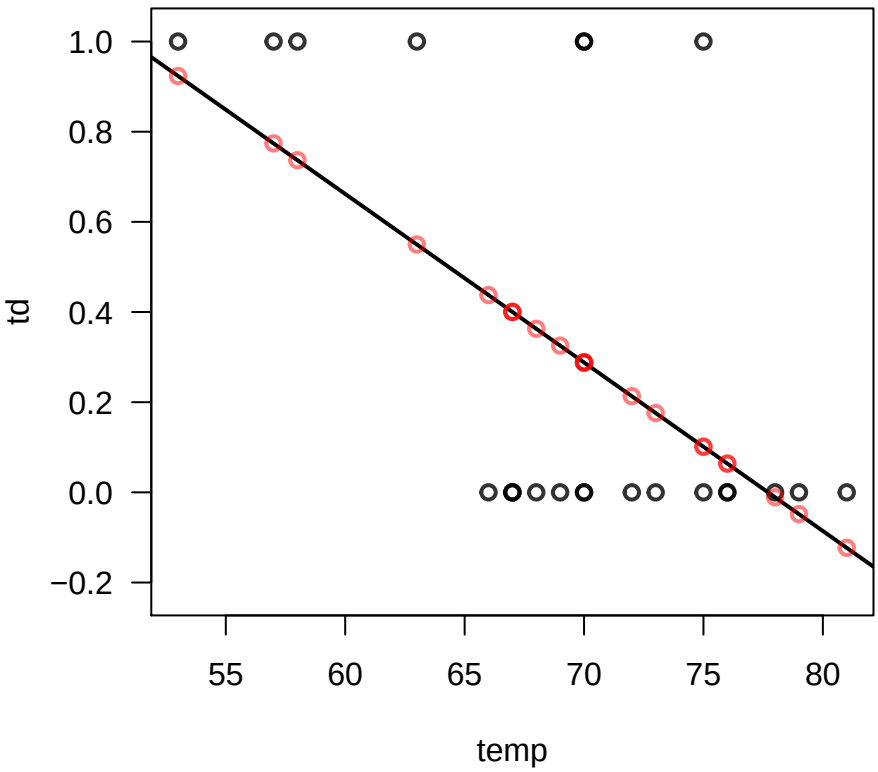

```
# Alternativ mit ggplot
#gg.shuttle <- ggplot(shuttle)
#grid.arrange(
#        gg.shuttle + geom_boxplot(aes(x = factor(td), y = temp)),
#        gg.shuttle + geom_violin(aes(x = factor(td), y = temp)),
#        nrow = 1)
```

→Vorsicht! Eine derartige Darstellung mag den Eindruck erwecken, dass es sich bei `temp` um die abhängige Variable handelt und bei `td` um die erklärende. In Hinblick auf evtl. Kausalität ist dies offensichtlicher Unsinn.

(b) **Fitten Sie nun mittels der Funktion `lm()` ein lineares Modell mit Temperatur in $^\circ F$ als Prädiktor. Erstellen Sie einen Plot, der die beobachteten Werte von `td` und `temp` zeigt, sowie die mit dem linearen Modell geschätzten Wahrscheinlichkeiten.**

```
# b)
# lineares Modell
lm.f <- lm(td ~ temp, data = shuttle)
# nun zur Verfuegung stehende Komponenten des lm-Objekts
names(lm.f)
```

```
##  [1] "coefficients"  "residuals"     "effects"       "rank"
##  [5] "fitted.values" "assign"        "qr"            "df.residual"
##  [9] "xlevels"       "call"          "terms"         "model"

# Streudiagramm td gegen temp
plot(td ~ temp, data = shuttle, lwd = 2,
        ylim = range(lm.f$fitted.values) + c(-0.1, 0.1),
        col = rgb(0,0,0, alpha = 0.8), las = 1)
# gefittete Werte eintragen
abline(lm.f$coef,lwd=2) # Spezialfall für lineares Modell,
# allgemeiner z.B. mit
points(shuttle$temp, lm.f$fitted.values, col = rgb(1,0,0, alpha = 0.5), lwd = 2)
```




```
# oder
## lines(temp,linMod$fitted.values,col=2,lwd=2) # bei nicht-geordneten temp nicht so gut
```

(c) **Welche Parameter-Schätzer ergeben sich, wenn die Temperatur in $°C$ gemessen wird? Welche Auswirkungen hätte die gleichzeitige Aufnahme von `temp` und `tempC` ins Modell?**

$$\begin{aligned}\eta &= \alpha_F + \beta_F \cdot T_F \\ &= \alpha_F + \beta_F(1.8T_C + 32) \\ &= \underbrace{\alpha_F + 32\beta_F}_{\alpha_C} + \underbrace{1.8\beta_F}_{\beta_C} \cdot T_C\end{aligned}$$

```
# c)
# per hand
beta0.c <- lm.f$coefficients[1] + 32 * lm.f$coefficients[2]
beta1.c <- 1.8 * lm.f$coefficients[2]
# lm fuer Celsius
```

```r
lm.c <- lm(td ~ tempC, data = shuttle)
# vergleich
c(beta0.c, beta1.c)

## (Intercept)        temp
##  1.70857143 -0.06728571

coefficients(lm.c)

## (Intercept)       tempC
##  1.70857143 -0.06728571
```

Sei $\hat{\theta}$ ML-Schätzer von θ und $\theta^* = h(\theta)$ eine *eineindeutige* Transformation und $\hat{\theta}^*$ der ML-Schätzer von $h(\theta)$. Dann gilt:

$$\hat{\theta}^* = h(\hat{\theta})$$

$$\Rightarrow \hat{\alpha}_C = \hat{\alpha}_F + 32 \cdot \hat{\beta}_F$$
$$\hat{\beta}_C = 1.8 \cdot \hat{\beta}_F$$

Gleichzeitige Aufnahme von `temp` und `tempC` ins Modell:

```r
# gleichzeitige Aufnahme von Temeratur in Fahrenheit und Celsius
summary(lm.multicol <- lm(td ~ temp + tempC, data = shuttle))

##
## Call:
## lm(formula = td ~ temp + tempC, data = shuttle)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -0.43762 -0.30679 -0.06381  0.17452  0.89881
##
## Coefficients: (1 not defined because of singularities)
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)  2.90476    0.84208   3.450  0.00240 **
## temp        -0.03738    0.01205  -3.103  0.00538 **
## tempC             NA         NA      NA       NA
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.3987 on 21 degrees of freedom
## Multiple R-squared:  0.3144,Adjusted R-squared:  0.2818
## F-statistic:  9.63 on 1 and 21 DF,  p-value: 0.005383
```

→perfekte Multikollinearität ⇒

I) Modell nicht eindeutig identifizierbar

II) keine lineare erwartungstreue Schätzung $\hat{\boldsymbol{\beta}}$ möglich (diese Problematik gilt nur im lin. Modell)

(d) **Betrachten Sie die `summary` des linearen Modells und interpretieren Sie die auftretenden Zahlen. Ist der lineare Term signifikant? Warum ist der hier verwendete Test problematisch? Was spricht darüber hinaus gegen die Verwendung des linearen Modells?**

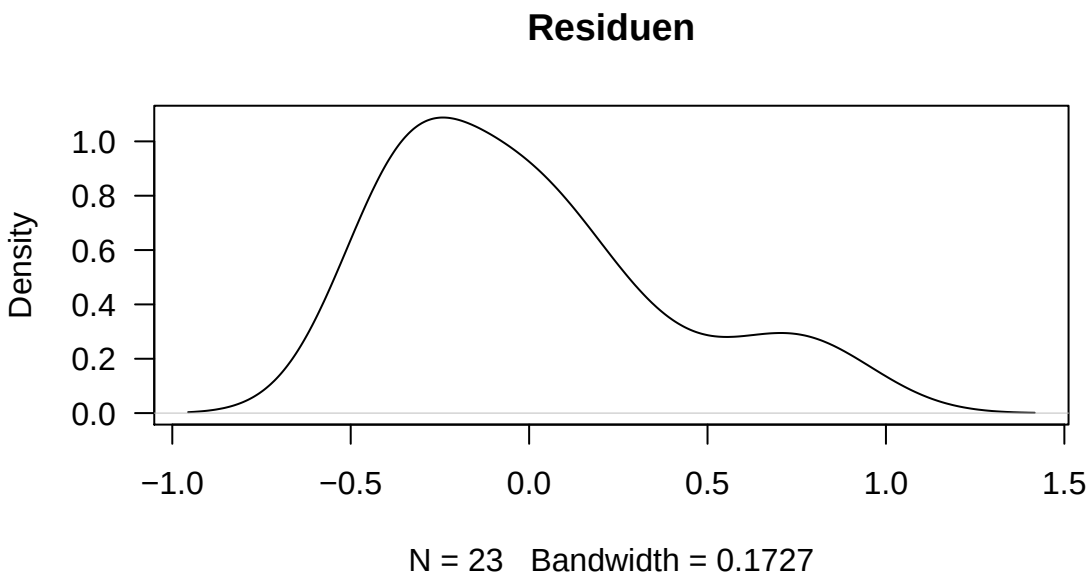
```r
# d)
# nochmals lineares modell fuer Fahrenheit
summary(lm.f)
```

```
##
## Call:
## lm(formula = td ~ temp, data = shuttle)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -0.43762 -0.30679 -0.06381  0.17452  0.89881
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)  2.90476    0.84208   3.450  0.00240 **
## temp        -0.03738    0.01205  -3.103  0.00538 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.3987 on 21 degrees of freedom
## Multiple R-squared:  0.3144,Adjusted R-squared:  0.2818
## F-statistic:  9.63 on 1 and 21 DF,  p-value: 0.005383
```

Interpretation:

- linkssteile Verteilung der Residuen (betrachte Quantile!), links von 0 offenbar mehr Masse als rechts

```r
## linkssteile Verteilung der Residuen
plot(density(lm.f$residuals), main = "Residuen", las = 1)
```



- beide Parameter signifikant von 0 verschieden (insb. linearer Term), aber Vorsicht (siehe unten)
- R^2 eher gering, aber Gesamtmodell als solches ist signifikant (ebenso Vorsicht)

```r
# R^2 eher gering, aber Gesamtmodell signifikant
anova(lm(td ~ 1, data = shuttle), lm.f)
```

```
## Analysis of Variance Table
##
## Model 1: td ~ 1
## Model 2: td ~ temp
##   Res.Df    RSS Df Sum of Sq      F   Pr(>F)
```

```
## 1     22 4.8696
## 2     21 3.3386  1     1.531 9.6301 0.005383 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

Linearer Term ist signifikant zum Niveau $\alpha = 0.01$. Aber t-Test verwendet Normalverteilung von $y =$ `td` (bzw. Approximation), und Normalerverteilung von ε bzw. y sicher nicht gegeben. (Für Approximation nicht genügend Daten vorhanden und Varianzhomogenität nicht erfüllt, siehe unten).

Gegen lineares Modell spricht z.B. (siehe auch Vorlesung):

- geschätzte Wahrscheinlichkeiten < 0 (und > 1) möglich
- keine Varianzhomogenität wegen $Var(y) = p(1-p)$ bei binärem y

zu Varianzhomogenität:
Annahme beim linearen Modell $Var(y|x) = \sigma^2$ unabhängig von x bzw. $E(y|x)$.
hier: binärer Response $y \in \{0, 1\}$ $\rightarrow Var(y|x) = p(x)(1-p(x))$
mit $p(x) = E(y|x) = P(y = 1|x)$ (hängt von Kovariablen und Parametern ab)

(e) **Fitten Sie nun mittels der Funktion `glm()` für binomialverteilten Response ein GLM mit Logit-, Probit- sowie komplementären Log-log-Link und betrachten Sie jeweils die `summary`. Erstellen Sie Plots analog zu (c). Welche Auswirkung hat der Übergang von $°C$ zu $°F$ (bzw. umgekehrt) im GLM (mit Begründung)?**

Modell:

$$E(y|x) = \pi(x) = h(\eta), \text{ mit } \eta = \alpha + \beta x$$

- Logit-Modell: $h(\eta) = \frac{\exp(\eta)}{1+\exp(\eta)}$
- Probit-Modell: $h(\eta) = \Phi(\eta)$
- Cloglog-Modell: $h(\eta) = 1 - exp(-exp(\eta))$

Motivation über latente Variable $\tilde{y}$ (stetig):

$$\tilde{y} = \mathbf{x}'\boldsymbol{\beta} + \varepsilon, \ \varepsilon \sim F$$

$\Rightarrow$ (normales lineares Modell)

$$y = 1, \text{ falls } \tilde{y} \geq \theta, \ y = 0 \text{ sonst}$$

$\rightarrow$

$$\pi = P(\mathbf{x}'\boldsymbol{\beta} + \varepsilon > 0) = 1 - h(-\mathbf{x}'\boldsymbol{\beta}) = h(\mathbf{x}'\boldsymbol{\beta})$$

```
# e)
# Logit-Modell: h(.) ist logistische Funktion
glm.logit <- glm(td ~ temp, data = shuttle, family=binomial)

# Probit-Modell: h(.) ist cdf von N(0,1)
glm.probit <- update(glm.logit, family = binomial(link = probit))

# Komplemetäres Log-log-Modell: h(.) ist cdf der Gompertz-Vtlg (nicht symmetrisch!)
```

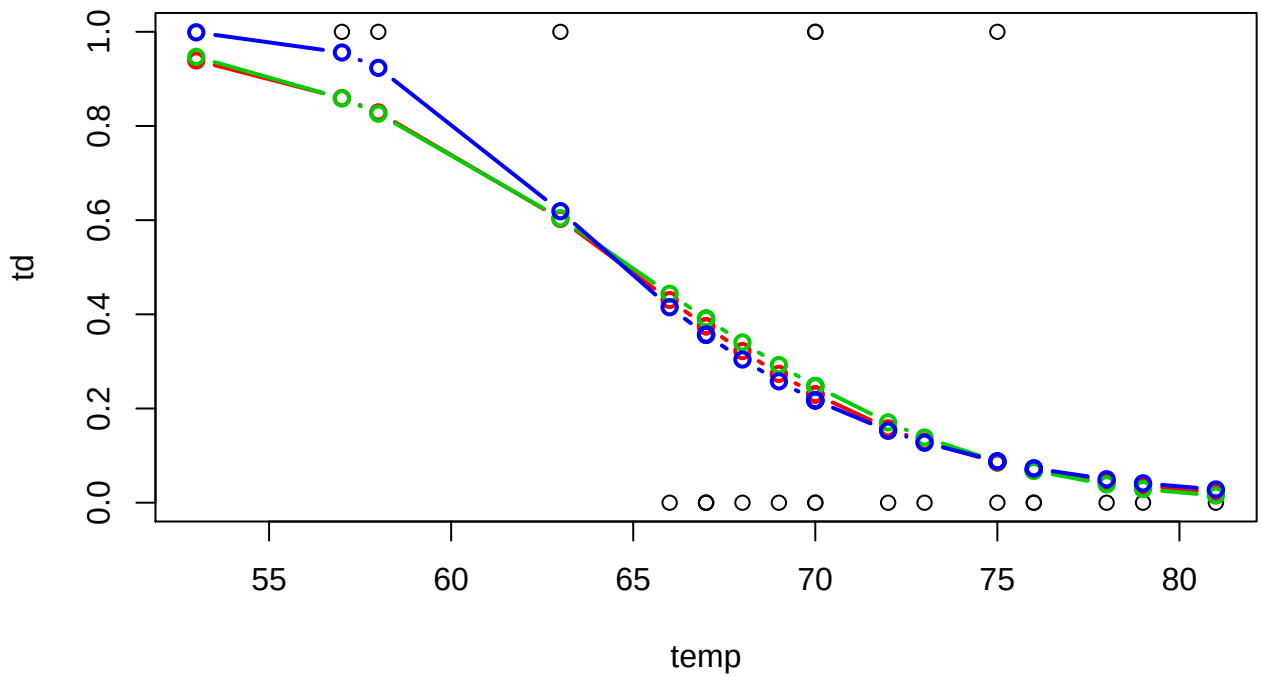
```r
glm.cloglog <- update(glm.logit, family = binomial(link = cloglog))

glm.list <- list(logit = glm.logit, probit = glm.probit, cloglog = glm.cloglog)

# Summaries
lapply(glm.list, summary)
```

```
## $logit
##
## Call:
## glm(formula = td ~ temp, family = binomial, data = shuttle)
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -1.0611  -0.7613  -0.3783   0.4524   2.2175
##
## Coefficients:
##             Estimate Std. Error z value Pr(>|z|)
## (Intercept)  15.0429     7.3786   2.039   0.0415 *
## temp         -0.2322     0.1082  -2.145   0.0320 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 28.267  on 22  degrees of freedom
## Residual deviance: 20.315  on 21  degrees of freedom
## AIC: 24.315
##
## Number of Fisher Scoring iterations: 5
##
##
## $probit
##
## Call:
## glm(formula = td ~ temp, family = binomial(link = probit), data = shuttle)
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -1.0831  -0.7930  -0.3747   0.4413   2.2081
##
## Coefficients:
##             Estimate Std. Error z value Pr(>|z|)
## (Intercept)  8.77490    3.87231   2.266   0.0234 *
## temp        -0.13510    0.05646  -2.393   0.0167 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 28.267  on 22  degrees of freedom
## Residual deviance: 20.378  on 21  degrees of freedom
## AIC: 24.378
##
## Number of Fisher Scoring iterations: 6
##
##
## $cloglog
##
## Call:
## glm(formula = td ~ temp, family = binomial(link = cloglog), data = shuttle)
##
```

```
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -1.0358  -0.7361  -0.3891   0.1729   2.2050
##
## Coefficients:
##             Estimate Std. Error z value Pr(>|z|)
## (Intercept) 12.30215    5.19483   2.368   0.0179 *
## temp        -0.19583    0.07809  -2.508   0.0122 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 28.267  on 22  degrees of freedom
## Residual deviance: 19.531  on 21  degrees of freedom
## AIC: 23.531
##
## Number of Fisher Scoring iterations: 8
```

Plots:

```
# Plots:
# daten
plot(td ~ temp, data = shuttle)
# geschaetzte Wahrscheinlichkeiten
t.index <- order(shuttle$temp)
lines(shuttle$temp[t.index], glm.logit$fitted.values[t.index],
        type = "b", lwd = 2, col = 2)
lines(shuttle$temp[t.index], glm.probit$fitted.values[t.index],
        type = "b", lwd = 2, col = 3)
lines(shuttle$temp[t.index],glm.cloglog$fitted.values[t.index],
        type = "b", lwd = 2, col = 4)
```




```
# Alternativ mit ggplot
#df.fitted <- sapply(glm.list, fitted)
#mdf.fitted <- melt(df.fitted)
#mdf.fitted$temp <- rep(shuttle$temp, ncol(df.fitted))
```

```
#gg.shuttle + geom_point(aes(x = temp, y = td))+
#        geom_line(data = mdf.fitted, aes(x = temp, y = value, colour = Var2)) +
#        geom_point(data = mdf.fitted, aes(x = temp, y = value, colour = Var2)) +
#        scale_color_discrete(name = "link")
```

Übergang von Fahrendheit zu Celsius:

$\hat{\alpha}$,$\hat{\beta}$ ML-Schätzer
$\Rightarrow$ Beim Übergang von $°C$ zu $°F$ Verhalten wie bei LM

(f) **Wie kann der Steigungsparameter im Logit-Modell interpretiert werden?**

$$\underbrace{\log}_{\text{Link-Funktion}}\left(\frac{\pi(x)}{1-\pi(x)}\right) = \underbrace{\alpha+\beta x}_{\text{lin. Prädiktor}} = \eta$$

$$\begin{aligned}\frac{\pi(x+1)}{1-\pi(x+1)} &= \exp(\alpha+\beta(x+1)) \\ &= \underbrace{\exp(\alpha+\beta x)}_{\pi(x)/(1-\pi(x))}\exp(\beta)\end{aligned}$$

(g) **Berechnen sie für alle drei Linkfunktionen die Wahrscheinlichkeit einer thermischen Überbeanspruchung bei der am Tage des Challenger-Unglücks herrschenden Temperatur von $31\,°F$ (*Hinweis:* die Response-Funktion des komplementären Log-log-Link ist die Verteilungsfunktion der Minimum-Extremwertverteilung (Gompertz-Verteilung), $h(\eta) = 1-\exp(-\exp(\eta))$).**

```
# von Hand:
# Logit-Modell
etalogit <- glm.logit$coef[1] + glm.logit$coef[2]*31 # Zugriff auf geschätzte Parameter
(pi.logit <- plogis(etalogit)) # plogis(): logistische Funktion

## (Intercept)
##   0.9996088

# Probit-Modell
eta.probit <- glm.probit$coef[1] + glm.probit$coef[2]*31
(pi.probit <- pnorm(eta.probit)) # pnorm(): cdf der N(0,1)-Vtlg.

## (Intercept)
##   0.9999978

# Komp. Log-log-Modell
eta.cloglog <- glm.cloglog$coef[1] + glm.cloglog$coef[2]*31
(pi.cloglog <- 1 - exp(-exp(eta.cloglog))) # vgl. Hinweis

## (Intercept)
##           1

## compare
c(pi.logit, pi.probit, pi.cloglog)

## (Intercept) (Intercept) (Intercept)
##   0.9996088   0.9999978   1.0000000
```

```r
# alternativ ueber predict funktion
# predict(model, newdata)
sapply(glm.list, predict, type = "response", newdata = data.frame(temp = 31))

##   logit.1  probit.1 cloglog.1
## 0.9996088 0.9999978 1.0000000

# Die Wahrscheinlichkeit für das Auftreten der therm. Überbeanspruchung bei
# einer Temperatur von 31°F ist in allen Modellen nahe eins; siehe auch
# Graphik aus e).
```

(h) **Für welche Temperatur beträgt diese Wahrscheinlichkeit jeweils 0.5?**

$$\pi(x) = h(\eta) = h(\alpha + \beta x) \Rightarrow \eta = h^{-1}(\pi) = g(\pi) \Rightarrow \alpha + \beta x = g(0.5) \Rightarrow x = \frac{g(0.5) - \alpha}{\beta}$$

- Logit:

$$\begin{aligned}
\pi = \frac{\exp \eta}{1 + \exp(\eta)} \quad &\Rightarrow \quad 1 - \pi = \frac{1}{1 + \exp(\eta)} \\
\Rightarrow \frac{\pi}{1 - \pi} &= \exp(\eta) \\
\Rightarrow \log\left(\frac{\pi}{1 - \pi}\right) &= \eta \\
\Rightarrow \log\left(\frac{0.5}{1 - 0.5}\right) &= g(0.5) = 0
\end{aligned}$$

- Probit:

$$\Phi^{-1}(\pi) = \eta \Rightarrow \Phi^{-1}(0.5) = 0$$

- Cloglog:

$$\begin{aligned}
\pi &= 1 - \exp(-\exp(\eta)) \\
&\Rightarrow \exp(-\exp(\eta)) = 1 - \pi \\
&\Rightarrow \eta = \log(-\log(1 - \pi)) = \log(-\log(1 - 0.5)) \neq 0
\end{aligned}$$

```r
# exakte Lösung, Herleitung der verwendeten Formeln siehe Übung
temp.logit <- -glm.logit$coef[1]/glm.logit$coef[2]
temp.probit <- -glm.probit$coef[1]/glm.probit$coef[2]
temp.cloglog <- (log(-log(0.5)) - glm.cloglog$coef[1])/glm.cloglog$coef[2]

c(temp.logit, temp.probit, temp.cloglog)

## (Intercept) (Intercept) (Intercept)
##    64.79464    64.95321    64.69111

## advanced
# liste mit linkfunktionen
link.functions <- list(
        logit = function(pi) log(pi/(1-pi)),
        probit = function(pi) qnorm(pi),
        cloglog = function(pi) log(-log(1 - pi)))

link.values <- sapply(link.functions, function(f) f(0.5))
link.values
```

```
##      logit     probit    cloglog
##  0.0000000  0.0000000 -0.3665129
```

```
coefs <- sapply(glm.list, coefficients)
coefs
```

```
##                  logit     probit    cloglog
## (Intercept) 15.0429016  8.7749032 12.3021525
## temp        -0.2321627 -0.1350958 -0.1958332
```

```
(link.values - coefs[1, ])/coefs[2, ]
```

```
##    logit   probit  cloglog
## 64.79464 64.95321 64.69111
```

```
## fuer Wkt von 0.9
(sapply(link.functions, function(f) f(0.9)) - coefs[1, ]) / coefs[2, ]
```

```
##    logit   probit  cloglog
## 55.33048 55.46697 58.56066
```