

Aufgabe 1

Laden Sie sich den Datensatz `madras.Rdata` von der Veranstaltungshomepage herunter. Der Datensatz wurde bereits in der Vorlesung besprochen.

- Laden Sie den Datensatz in R. Erzeugen Sie einen neuen Datensatz, der ausschließlich Patienten mit vollständigen Beobachtungen enthält, also Patienten bei denen für alle 12 Monate eine Beobachtung vorliegt.
- Visualisieren Sie die relative Häufigkeit von Schüben pro Monat in Abhängigkeit des Geschlechts.
- Schreiben Sie ein Modell für die Wahrscheinlichkeit einen Schub zu erleiden in Abhängigkeit von Monat, Geschlecht, sowie einem zufälligen Intercept pro Person auf und fitten Sie dieses in R.
Hinweis: Verwenden Sie die Funktion `glmer()` aus dem R-Paket `lme4`.
- Berechnen Sie für das Modell aus (c) das marginale Odds Ratio für Geschlecht. Wie ist das konditionale Odds Ratio im Vergleich? Interpretieren Sie konditionales und marginales Odds Ratio.
- Fügen Sie dem Modell aus (c) den Interaktionsterm aus Geschlecht und Monat hinzu.
- Fügen Sie dem Modell aus (c) einen Random Slope pro Person hinzu. Schätzen Sie das Modell einmal unter der Annahme, dass random intercept und random slope unabhängig sind und einmal mit Abhängigkeit.
- Erstellen Sie für die Modelle aus den Teilaufgaben (c), (e) und (f) Graphiken der vorhergesagten Wahrscheinlichkeiten für die Personen. Vergleichen Sie die Verläufe der Wahrscheinlichkeiten.
- Zusatzaufgabe:* Erstellen Sie eine Graphik zur Interpretation der festen Effekte abhängig von den Quantilen der zufälligen Effekte.
- Erweitern Sie das Modell aus (c) um die Interaktion von Geschlecht und Monat und berechnen Sie die Konfidenzintervalle für das Modell basierend auf der Profile-Loglikelihood. Entscheiden Sie anhand des Konfidenzintervalls, ob die Interaktion von Geschlecht und Monat im Modell verbleiben soll.
Hinweis: Siehe `?confint.merMod`.

Lösung 1

```
(a) library(RColorBrewer)
     palette(brewer.pal(9, 'Set1'))
     library(ggplot2)

##### load data
u <- paste0("https://www.elab.moodle.elearning.lmu.de/pluginfile.php/32320/",
            "course/section/12446/madras.csv")
```

```

madras <- read.csv(u)
str(madras, 1)

'data.frame': 922 obs. of 10 variables:
 $ X      : int  1 2 3 4 5 6 7 8 9 10 ...
 $ id     : int  1 1 1 1 1 1 1 1 1 1 ...
 $ y      : int  1 1 1 1 1 0 0 0 0 0 ...
 $ month  : int  0 1 2 3 4 5 6 7 8 9 ...
 $ age    : int  0 0 0 0 0 0 0 0 0 0 ...
 $ gender : Factor w/ 2 levels "Female","Male": 2 2 2 2 2 2 2 2 2 2 ...
 $ monthXage : int  0 0 0 0 0 0 0 0 0 0 ...
 $ monthXgender: int  0 0 0 0 0 0 0 0 0 0 ...
 $ initial : int  0 1 0 0 0 0 0 0 0 0 ...
 $ count  : int  1 2 3 4 5 5 5 5 5 5 ...

# madrasC: only keep patients with all 12 observations
completeId <- names(table(madras$id)[table(madras$id)==12])
madrasC <- subset(madras, madras$id %in% completeId)
madrasC$id <- factor(madrasC$id) # drop unused levels

# control
table(madrasC$id)

 1  2  4  5  6  7  8  9 10 11 13 14 15 16 18 20 21 22 23 24 25 26 27 29 30
12 12 12 12 12 12 12 12 12 12 12 12 12 12 12 12 12 12 12 12 12 12 12 12
31 32 33 34 35 36 37 38 39 41 42 43 44 45 46 47 48 50 51 52 53 54 55 56 57
12 12 12 12 12 12 12 12 12 12 12 12 12 12 12 12 12 12 12 12 12 12 12 12
59 60 61 62 64 65 66 67 68 70 71 72 73 74 77 80 81 85 90
12 12 12 12 12 12 12 12 12 12 12 12 12 12 12 12 12 12 12

```

```

(b) # calculate relative frequency of attacks over time for males and females
yMatMales <- matrix(subset(madrasC$y, madrasC$gender=="Male"), nrow=12)
relFMales <- apply(yMatMales, 1, mean)

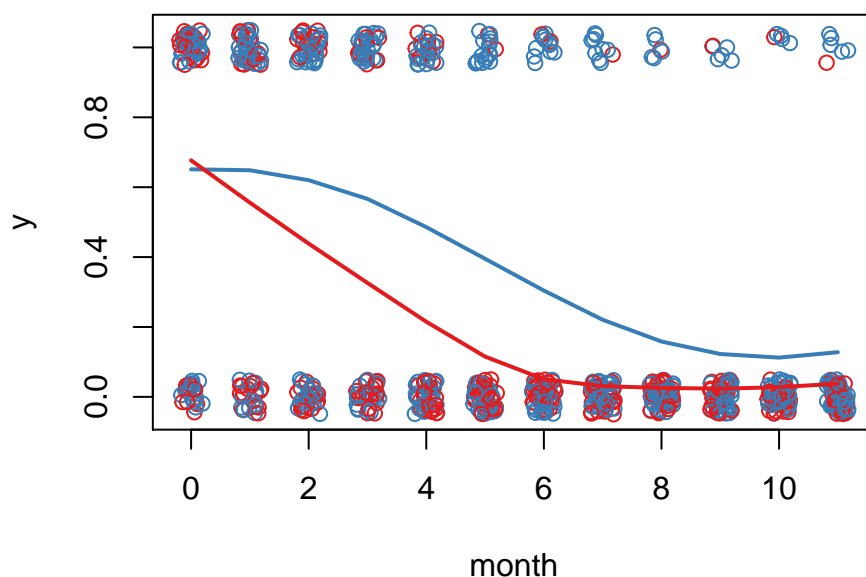
yMatFemales <- matrix(subset(madrasC$y, madrasC$gender=="Female"), nrow=12)
relFFemales <- apply(yMatFemales, 1, mean)

# add random noise to response and to month before plotting
madrasC$yN <- madrasC$y + runif(length(madrasC$y), -0.05, 0.05)
madrasC$monthN <- madrasC$month + runif(length(madrasC$month), -0.2, 0.2)

```

```
with(madrasC, plot(yN ~ monthN, col=gender,
                  main="rel. Haeufigkeit der Schuebe ueber die Zeit",
                  ylab="y", xlab="month"))
m <- 0:11
lines(m, predict(loess(relFMales~m)), col=2, lwd=2)
lines(m, predict(loess(relFFemales~m)), col=1, lwd=2)
```

rel. Haeufigkeit der Schuebe ueber die Zeit



(c) GLMM für

$$y_{it} = \begin{cases} 1, & \text{PatientIn } i \text{ erleidet Schub in Monat } t \\ 0, & \text{sonst.} \end{cases}$$

- *Verteilungsannahme für Zielgröße:*

$$y_{it} | \mathbf{x}_i, b_i \sim \text{Bin}(0, \pi_{it}), \text{ bedingt unabhängig verteilt}$$

- *Strukturannahme:*

Modell mit zufälligen Intercepts und festen Effekten von Monat und Geschlecht:

$$\pi_{it} = P(y_{it} | \mathbf{x}_i, b_i) = g^{-1}(\eta_{it}) = g^{-1}(\beta_0 + \beta_1 \text{month}_{it} + \beta_2 \text{sex}_i + b_i), \quad i = 1, \dots, 69, \quad t = 0, \dots, 11,$$

mit $g^{-1}(\eta) = \frac{\exp(\eta)}{1 + \exp(\eta)}$ inverse Linkfunktion, $\text{sex}_i = 1$ für Frauen und 0 für Männer.

- *Verteilungsannahme für zufällige Effekte:* $b_i \stackrel{\text{iid}}{\sim} N(0, \tau^2)$

```
library(lme4)
?glmer
```

```

m1 <- glmer(y ~ month + gender + (1|id), family=binomial(), data=madrasC)
# summary(m1)

fixef(m1)

(Intercept)      month  genderMale
  0.3590179  -0.5731068   1.5735114

exp(fixef(m1))

(Intercept)      month  genderMale
  1.4319224   0.5637712   4.8235560

#ranef(m1)

summary(m1)$varcor

Groups Name      Std.Dev.
id      (Intercept) 2.2154

```

(d) Konditionales Odds Ratio für Geschlecht für Individuen (vgl. Vorlesung):

$$\frac{P(y_{it} = 1|x_i = 1, b_i)}{P(y_{it} = 0|x_i = 1, b_i)} \bigg/ \frac{P(y_{it} = 1|x_i = 0, b_i)}{P(y_{it} = 0|x_i = 0, b_i)} = \frac{\exp(\beta_0 + \beta_1 t + \beta_2 \cdot 1 + b_i)}{\exp(\beta_0 + \beta_1 t + \beta_2 \cdot 0 + b_i)} = \exp(\beta_2)$$

Marginales Odds Ratio für Geschlecht für Population (vgl. Vorlesung):

$$\frac{P(y_{it} = 1|x_i = 1)}{P(y_{it} = 0|x_i = 1)} \bigg/ \frac{P(y_{it} = 1|x_i = 0)}{P(y_{it} = 0|x_i = 0)} = \frac{\mathbb{E}_{b_i}[P(y_{it} = 1|x_i = 1, b_i)]}{\mathbb{E}_{b_i}[P(y_{it} = 0|x_i = 1, b_i)]} \bigg/ \frac{\mathbb{E}_{b_i}[P(y_{it} = 1|x_i = 0, b_i)]}{\mathbb{E}_{b_i}[P(y_{it} = 0|x_i = 0, b_i)]} \neq \exp(\beta_2)$$

Berechnung in R:

- Konditionale Odds Ratio für **sex**

```

exp(fixef(m1)[3])

genderMale
  4.823556

```

- Approximiere Marginale Odds Ratio (Möglichkeit 1): Simuliere Daten aus dem Modell um konditionalen Erwartungswert zu schätzen (Code von Lisa Steyer und Tore Erdmann)

```

## E_b(P(y=1|x,b))
pr_y1 <- function(x, beta, b) {
  mean(exp(x %*% beta + b) / (1 + exp(x %*% beta + b)))
}

```

```

}

## E_b(P(y=0|x,b))
pr_y0 <- function(x, beta, b) {
  mean( 1 / (1 + exp(x %*% beta + b)))
}

# get estimated sd of random intercepts
tau <- sqrt(summary(m1)$varcor$id[1])

# ## estimate sd of random intercepts by using the predictions of the b_i s
# ## smaller than direct estimate! (gives a result closer to solution 2)
# tau <- sd(ranef(m1)[[1]][[1]])

## compute Odds ratios at monthi
# n number of simulated observations from the model
# fixef(m1): beta_0, beta_month, beta_sex
# for men: x=c(1, monthti, 1)
# for women: x=c(1, monthti, 0)
f1 <- function(monthi, n = 10000) {
  (pr_y1(x = c(1, monthi, 1), beta = fixef(m1),
    b = rnorm(n = n, sd = tau)) /
    pr_y0(x = c(1, monthi, 1), beta = fixef(m1),
    b = rnorm(n = n, sd = tau)) ) /
  (pr_y1(x = c(1, monthi, 0), beta = fixef(m1),
    b = rnorm(n = n, sd = tau)) /
    pr_y0(x = c(1, monthi, 0), beta = fixef(m1),
    b = rnorm(n = n, sd = tau)) )
}

set.seed(42)
# apply function f1() to the month 0:11
(margORtime1 <- sapply(0:11, function(i) f1(i, n = 10000)))

[1] 2.487786 2.482934 2.469785 2.474483 2.501988 2.572074 2.626738
[8] 2.723071 2.821254 3.024920 3.162011 3.410914

#plot(x = 0:11, y = margORtime1)

## mean overall odds ratio
(margOR1 <- mean(margORtime1))

```

```
[1] 2.72983
```

- Approximiere Marginale Odds Ratio (Möglichkeit 2): Verwende Vorhersagen b_i um konditionalen Erwartungswert zu schätzen:

```
# is the person female?
female <-
  madrasC$gender[seq(1, by = 12, l = length(table(madrasC$id)))] == "Female"

# random effects females
bFemale <- ranef(m1)$id$(Intercept)[female]
# random effects males
bMale <- ranef(m1)$id$(Intercept)[!female]

# get the fixed effects
time <- 0:11
beta0 <- fixef(m1)[1]
beta1 <- fixef(m1)[2]
beta2 <- fixef(m1)[3]

# probabilities of attacks for femals depending on b and month
# one woman per column, time in rows
pFemale <- sapply(bFemale, function(b){

  exp(beta0 + beta1 * time + b) /
    (1 + exp(beta0 + beta1 * time + b))

})

# probabilities of attacks for males depending on b and month
# one man per column, time in rows
pMale <- sapply(bMale, function(b){

  exp(beta0 + beta1 * time + beta2 + b) /
    (1 + exp(beta0 + beta1 * time + beta2 + b))

})

# odds ratios for time=0,...,11 - expectation is
# approximated by the mean over all b
(margORtime2 <- ( rowMeans(pMale) /rowMeans(1-pMale) ) /
```

```

( rowMeans(pFemale) /rowMeans(1-pFemale) ))

[1] 2.260081 2.276595 2.341029 2.460902 2.645851 2.908924 3.269907
[8] 3.759219 4.417482 5.282013 6.353983 7.564167

# mean marginal odds ratio over all b and
# times = approximation of expectation
(margOR2 <- mean(( rowMeans(pMale) /rowMeans(1-pMale)) /
                 ( rowMeans(pFemale) /rowMeans(1-pFemale))))

[1] 3.795013

```

- Vergleich beider Approximationen

```

#####
# compare with conditional odds ratio of sex
exp(fixef(m1)[3])

genderMale
  4.823556

margOR1

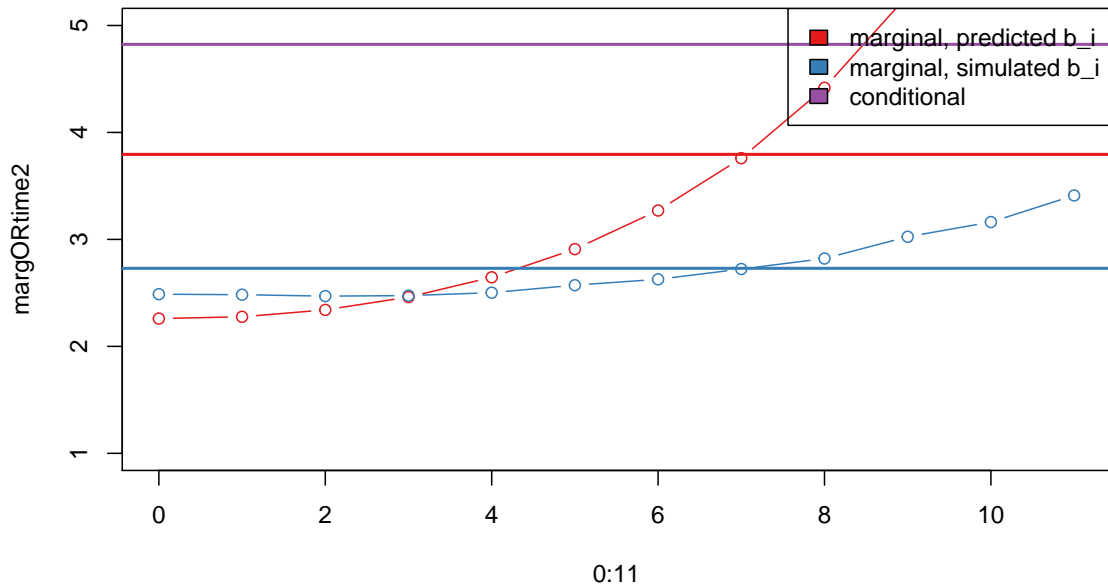
[1] 2.72983

margOR2

[1] 3.795013

##### plot marginal and conditional odds ratio for comparison
plot(0:11, margORtime2, type="b", col=1, ylim=c(1,5))
lines(x = 0:11, y = margORtime1, type="b", col=2)
abline(h=exp(beta2), col=4, lwd=2)
abline(h=mean(margORtime2), col=1, lwd=2)
abline(h=mean(margORtime1), col=2, lwd=2)
legend("topright", fill=c(c(1,2,4)),
       legend=c("marginal, predicted b_i",
                "marginal, simulated b_i",
                "conditional" ) )

```



```

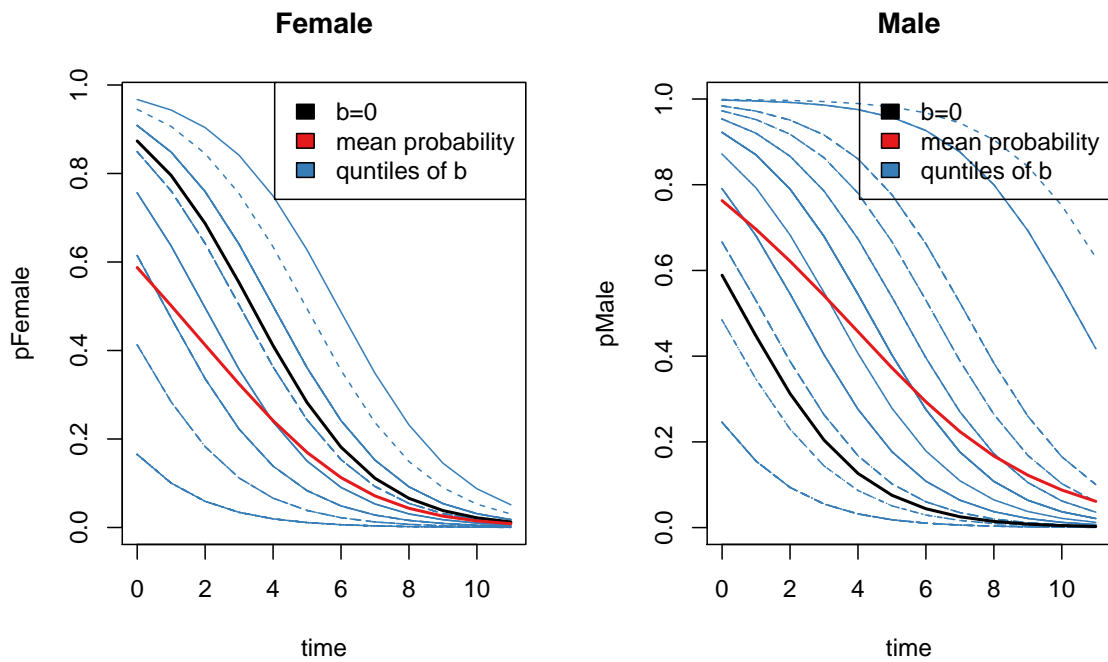
### comment: if prior interest is in population parameter:
### fit a generalized estimating equation model (GEE), R-package gee

#####
# plot probabilities over time for males and females
par(mfrow=c(1,2))

matplot(time, pFemale, type="l", main="Female", col=2)
# probability for b=0 (black)
lines(time, exp(beta0 + betal * time + beta2) /
      (1 + exp(beta0 + betal * time + beta2)), lwd=2)
# mean probability (red)
lines(time, apply(pFemale, 1, mean), col = 1, lwd = 2)
legend("topright", legend=c("b=0","mean probability", "quantiles of b"),
      fill=c("black", 1, 2))

matplot(time, pMale, type="l", main="Male", col=2)
lines(time, exp(beta0 + betal * time) /
      (1 + exp(beta0 + betal * time)), lwd=2)
lines(time, apply(pMale, 1, mean), col = 1, lwd = 2)
legend("topright", legend=c("b=0","mean probability", "quantiles of b"),
      fill=c("black", 1, 2))

```

Interpretation konditionales OR:

Die Chance, dass eine bestimmter Mann mit dem gleichen zufälligen Effekt wie eine bestimmte Frau, einen Anfall erleidet, ist um den Faktor 4.823556 höher, wenn alle anderen Kovariablen gleich sind.

Interpretation marginales OR:

Die Chance, dass ein Mann einen Anfall erleidet, ist um den Faktor 2.7298299 höher, als für eine Frau, wenn alle anderen Kovariablen gleich sind.

Marginales Odds Ratio ≈ 2.7298299 und konditionales Odds Ratio 4.823556.

Das marginale Odds Ratio ist näher an 1 als das konditionale Odds Ratio. Das bedeutet, dass der marginale Effekt weniger extrem ist, als der konditionale. Das heißt der Effekt auf Populationsebene ist weniger extrem als der Effekt auf individueller Ebene.

Kommentar: Wenn das Hauptinteresse auf dem Populationsparameter liegt: fitte ein generalized estimating equation model (GEE)

- GEE schätzt Populationseffekte
- GLMM schätzt individuelle Effekte

```
(e) m2 <- update(m1, .~. + month:gender)
# summary(m2)
```

```
(f) # without correlation of random effects
m3 <- update(m2, .~. + (-1 + month|id))
# summary(m3)

# with correlation of random effects
m4 <- update(m2, .~. - (1|id) + (month|id))
```

```
# summary(m4)
```

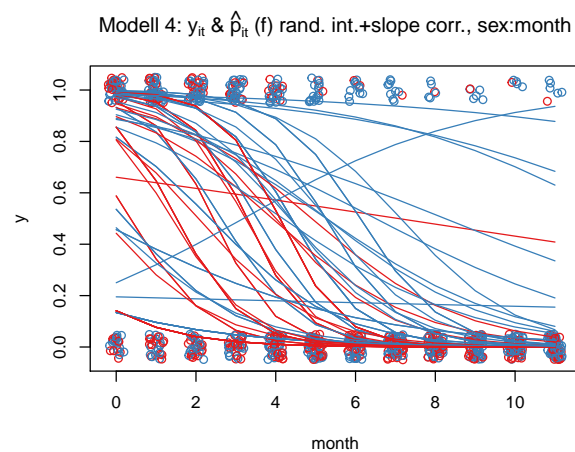
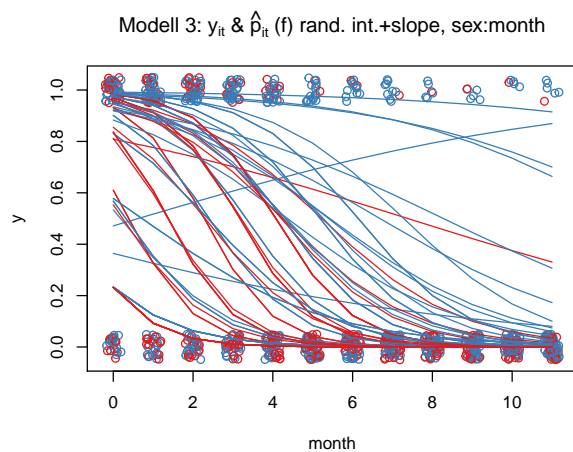
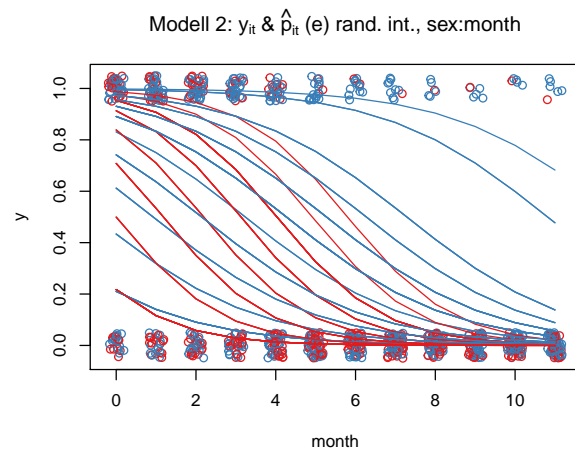
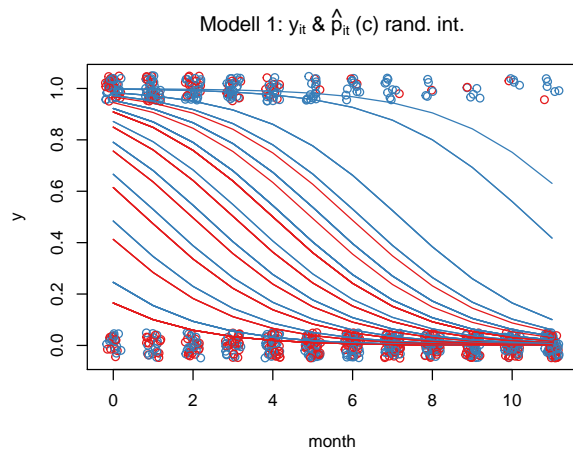
```
(g) par(mfrow=c(2,2))
```

```
with(madrasC, plot(yN ~ monthN, col=gender,
                  main=bquote(paste("Modell 1: ", y[it]," & ",
                                    hat(p)[it], " (c) rand. int.")),
                  ylab="y", xlab="month"))
fitted1mat <- matrix(fitted(m1), nrow=12)
matplot(0:11, fitted1mat, type="l", lty=1,
        col=madrasC$gender[seq(1,by=12,l=69)], add=TRUE)

with(madrasC, plot(yN ~ monthN, col=gender,
                  main=bquote(paste("Modell 2: ", y[it]," & ",
                                    hat(p)[it], " (e) rand. int., sex:month")),
                  ylab="y", xlab="month"))
fitted2mat <- matrix(fitted(m2), nrow=12)
matplot(0:11, fitted2mat, type="l", lty=1,
        col=madrasC$gender[seq(1,by=12,l=69)], add=TRUE)

with(madrasC, plot(yN ~ monthN, col=gender,
                  main=bquote(paste("Modell 3: ", y[it]," & ",
                                    hat(p)[it], " (f) rand. int.+slope, sex:month")),
                  ylab="y", xlab="month"))
fitted3mat <- matrix(fitted(m3), nrow=12)
matplot(0:11, fitted3mat, type="l", lty=1,
        col=madrasC$gender[seq(1,by=12,l=69)], add=TRUE)

with(madrasC, plot(yN ~ monthN, col=gender,
                  main=bquote(paste("Modell 4: ", y[it]," & ",
                                    hat(p)[it], " (f) rand. int.+slope corr., sex:month")),
                  ylab="y", xlab="month"))
fitted4mat <- matrix(fitted(m4), nrow=12)
matplot(0:11, fitted4mat, type="l", lty=1,
        col=madrasC$gender[seq(1,by=12,l=69)], add=TRUE)
```



```
(h) # look at random effects of m1
quantile(exp(ranef(m1)$id$('Intercept)'), p=c(.2, .4, .6, .8))

      20%      40%      60%      80%
0.1380641 0.6336754 2.1663892 5.8273982

# summary(m1)

beta0 <- fixef(m1)[1]
beta1 <- fixef(m1)[2]

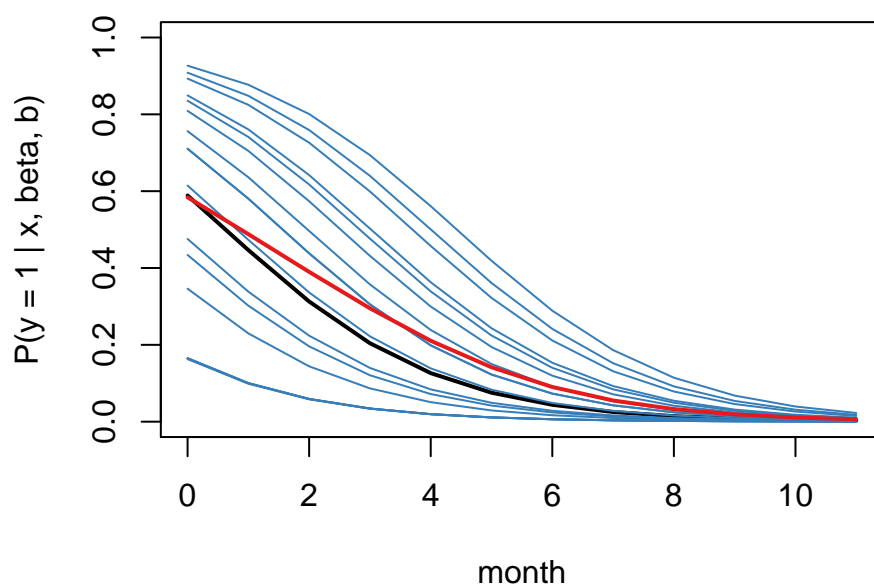
# quantiles of "exp(ranef)"
expquants <- quantile(exp(ranef(m1)$id$('Intercept)'), p=seq(0.1, 0.9, 0.05))
time <- 0:11

# probability depending on quantile of b
indv <- sapply(expquants, function(q){q * exp(beta0 + beta1 * time) /
      (1 + q * exp(beta0 + beta1 * time))})
```

```

plot(time, exp(beta0 + beta1 * time) / (1 + exp(beta0 + beta1 * time)),
     type = 'l', ylim = c(0:1), lwd = 2,
     ylab = 'P(y = 1 | x, beta, b)', xlab = 'month')
for (q in 1:length(expquants)){
  lines(time, indiv[, q], type = 'l', col = 2)
}
lines(time, apply(indv, 1, mean), col = 1, lwd = 2)

```



```

(i) m5 <- glmer(y ~ month + gender + month:gender + (1|id),
               family=binomial(), data=madrasC)
# could also use the update-function

# summary(m5)

# Get confidence intervals for coefficients using the profile-likelihood
# can save computation time, e.g. when looking at different levels
prL <- profile(m5)
confint(prL, level = 0.95)

```

	2.5 %	97.5 %
.sig01	1.69707372	2.9094726
(Intercept)	-0.14971604	1.9503384

```
month          -0.95457716 -0.5819813
genderMale     -0.70672067  2.0802320
month:genderMale 0.06397941  0.4761822
```

or more easily, use confint() directly on model-object

```
confint(m5, level = 0.95, method = "profile")
```

```
                2.5 %    97.5 %
.sig01          1.69707372  2.9094726
(Intercept)     -0.14971604  1.9503384
month           -0.95457716 -0.5819813
genderMale      -0.70672067  2.0802320
month:genderMale 0.06397941  0.4761822
```