Unit 3

Regularized generalized linear models

Generalized linear models

Introduction

- Linear models are well suited for regression analyses when the response variable is continuous and at least approximately normal.
- In many applications the response is not a continuous variable, but rather binary, categorical, or a count variable.
- Difficulties with linear regression models are also encountered for continuous variables with limited support and which are considerably skewed.
- Generalized linear models (GLMs) unify many regression approaches with response variables that do not necessarily follow a normal distribution.

Binary regression

Binary regression

- Assume that (ungrouped) data on N objects or individuals are given in the form $(y_i, x_{i1}, \ldots, x_{ik})$, $i = 1, \ldots, N$, with the binary response y coded by 0 and 1 and covariates denoted by $x_1, \ldots, x_k$.
- The covariates $x_1, \ldots, x_k$ may have been derived from an appropriate transformation or coding of the original variables.
- The main goal of a binary regression analysis is then to model and estimate the effects of the covariates on the (conditional) probability

$$\pi_i = \mathrm{P}(y_i = 1 | \boldsymbol{x}_i) = \mathrm{E}(y_i | \boldsymbol{x}_i),$$

  for the outcome $y_i = 1$ and given values of the covariates $\boldsymbol{x}_i = (x_{i1}, \ldots, x_{ik})^\top$.
- The response variables are assumed to be (conditionally) independent.

Linear probability model

The **linear probability model** is given by

$$\pi_i = \beta_0 + \beta_1 x_{i1} + \ldots + \beta_k x_{ik},$$

with linear predictor

$$\eta_i = \beta_0 + \beta_1 x_{i1} + \ldots + \beta_k x_{ik} = \boldsymbol{x}_i^\top \boldsymbol{\beta},$$

with $\boldsymbol{\beta} = (\beta_0, \beta_1, \ldots, \beta_k)^\top$ and $\boldsymbol{x}_i = (1, x_{i1}, \ldots, x_{ik})^\top$.

- The linear predictor is equal to the success probability.
- The linear predictor must lie in the interval $[0, 1]$ for all vectors $\boldsymbol{x}$.
- This requires restrictions on the parameters $\boldsymbol{\beta}$ that are difficult to handle in the estimation process.

Binary regression models

- Combine the probability π_i with the linear predictor η_i through a relation of the form

$$\pi_i = h(\eta_i) = h(\beta_0 + \beta_1 x_{i1} + \ldots + \beta_k x_{ik}),$$

  where h is a strictly monotonically increasing cumulative distribution function on the real line.
- This ensures $h(\eta) \in [0, 1]$.
- In addition one can also write

$$\eta_i = g(\pi_i),$$

  with the inverse function $g = h^{-1}$.
- Within the framework of GLMs, h is called the **response function** and $g = h^{-1}$ is known as the **link function**.
- Logit and probit models are the most widely used binary regression models.

Binary regression models / 2

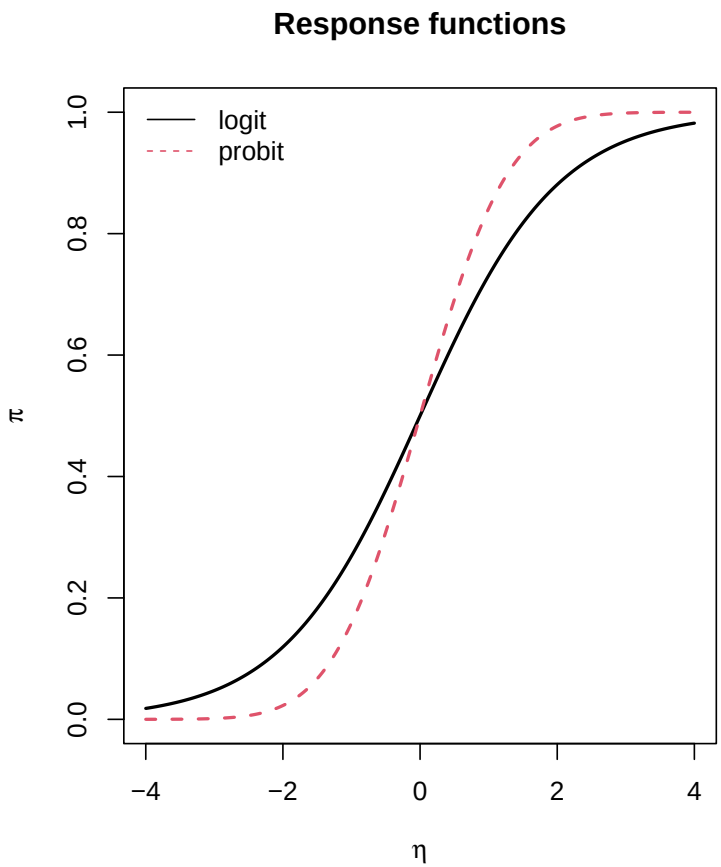
Logit model:

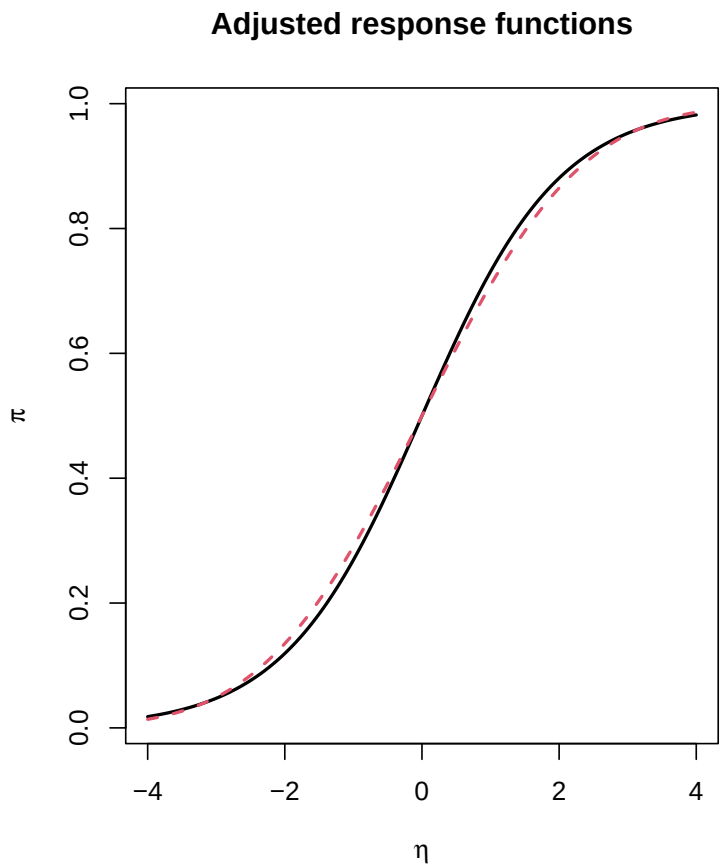
$$\pi = \frac{\exp(\eta)}{1 + exp(\eta)} \qquad \Leftrightarrow \qquad \log \frac{\pi}{1 - \pi} = \eta.$$

Probit model:

$$\pi = \Phi(\eta) \qquad \Leftrightarrow \qquad \phi^{-1}(\pi) = \eta.$$

Binary regression models / 3





Binary regression models / 4

- Instead of the probit one could have used the more general cumulative function h of a $N(0, \sigma^2)$ distribution with any choice of variance $\sigma^2 \neq 1$.
- Standardizing h yields the relation

$$\pi(\eta) = h(\boldsymbol{x}^\top \boldsymbol{\beta}) = \Phi(\boldsymbol{x}^\top \boldsymbol{\beta}/\sigma) = \Phi(\boldsymbol{x}^\top \tilde{\boldsymbol{\beta}}),$$

  where $\tilde{\boldsymbol{\beta}} = \boldsymbol{\beta}/\sigma$.
- The resulting model for the probability $\pi(\eta)$ based on $h(\eta)$ with $\eta = \boldsymbol{x}^\top \boldsymbol{\beta}$ is equivalent to a probit model with the rescaled parameters $\tilde{\boldsymbol{\beta}} = \boldsymbol{\beta}/\sigma$.

Binary regression models / 5

- For an adequate comparison of the models, the mean and variance of the CDFs need to be matched.
- The logistic distribution function has variance $\pi^2/3$. Thus it needs to be compared to a rescaled normal distribution whose variance is adjusted to $\sigma^2 = \pi^2/3$.
- The logit and adjusted probit response functions are very similar.
- The estimated coefficients of a logit model differ from the corresponding values of a probit model (with $\sigma^2 = 1$) approximately by the factor $\sigma = \pi/\sqrt{3}$.
- The estimated probabilities $\pi(\eta)$ are very similar.
- This indicates that instead of the absolute values of the (estimated) coefficients rather the ratios should be interpreted.

Binary models and latent linear models

- Binary regression models can be derived by considering a **latent (unobserved) continuous response variable**.
- The latent variable is connected with the observed binary response via a threshold mechanism.
- Suppose we are investigating the decision of some individuals $i = 1, \ldots, N$ when choosing between two alternatives $y = 0$ and $y = 1$.
- Assume that individuals assign utilities u_{i0} and u_{i1} to each of the two alternatives.
- The alternative that maximizes the utility is chosen, i.e.,

$$y_i = \begin{cases} 1 & u_{i1} > u_{i0}, \\ 0 & u_{i1} \leq u_{i0}. \end{cases}$$

Binary models and latent linear models / 2

- Assuming that the unobserved utilities can be additively decomposed and follow a linear model, we obtain

$$u_{i1} = \boldsymbol{x}_i^\top \tilde{\boldsymbol{\beta}}_1 + \tilde{\epsilon}_{i1},$$
$$u_{i0} = \boldsymbol{x}_i^\top \tilde{\boldsymbol{\beta}}_0 + \tilde{\epsilon}_{i0},$$

  with $\boldsymbol{x}_i = (1, x_{i1}, \ldots, x_{ik})^\top$.
- The unknown coefficient vectors $\tilde{\boldsymbol{\beta}}_1$ and $\tilde{\boldsymbol{\beta}}_0$ determine the effect of the explanatory variables on the utilities.
- The "errors" $\tilde{\epsilon}_{i1}$ and $\tilde{\epsilon}_{i0}$ include the effects of unobserved explanatory variables.

Binary models and latent linear models / 3

- Equivalently, we may choose to investigate utility differences

$$\tilde{y}_i = u_{i1} - u_{i0} = \boldsymbol{x}_i^\top(\tilde{\boldsymbol{\beta}}_1 - \tilde{\boldsymbol{\beta}}_0) + \tilde{\epsilon}_{i1} - \tilde{\epsilon}_{i0} = \boldsymbol{x}_i^\top \boldsymbol{\beta} + \epsilon_i,$$

  with $\boldsymbol{\beta} = \tilde{\boldsymbol{\beta}}_1 - \tilde{\boldsymbol{\beta}}_0$ and $\epsilon_i = \tilde{\epsilon}_{i1} - \tilde{\epsilon}_{i0}$.

- Based on this framework, the binary responses y_i follow a Bernoulli distribution with

$$\begin{aligned}\pi_i &= \mathrm{P}(y_i = 1|\boldsymbol{x}_i) = \mathrm{P}(\tilde{y}_i > 0|\boldsymbol{x}_i) = \mathrm{P}(\boldsymbol{x}_i^\top \boldsymbol{\beta} + \epsilon_i > 0) \\ &= \int I(\boldsymbol{x}_i^\top \boldsymbol{\beta} + \epsilon_i > 0) f(\epsilon_i) d\epsilon_i,\end{aligned}$$

  where $I(\cdot)$ is the indicator function and f is the probability density of ϵ_i.

Binary models and latent linear models / 4

- We obtain different models depending on the choice of f:
  - When ϵ_i follows a logistic distribution, we obtain the logit model.
  - When ϵ_i follows a standard normal distribution, we obtain the probit model.

Interpretation of the logit model

Based on the linear predictor

$$\eta_i = \beta_0 + \beta_1 x_{i1} + \ldots + \beta_k x_{ik} = \boldsymbol{x}_i^\top \boldsymbol{\beta},$$

the odds

$$\frac{\pi_i}{1 - \pi_i} = \frac{\mathrm{P}(y_i = 1 | \boldsymbol{x}_i)}{\mathrm{P}(y_i = 0 | \boldsymbol{x}_i)}$$

follow the multiplicative model

$$\frac{\mathrm{P}(y_i = 1 | \boldsymbol{x}_i)}{\mathrm{P}(y_i = 0 | \boldsymbol{x}_i)} = \exp(\beta_0) \cdot \exp(\beta_1 x_{i1}) \cdot \ldots \cdot \exp(\beta_k x_{ik}).$$

Interpretation of the logit model / 2

If x_{i1} increases by 1 unit to $x_{i1} + 1$, the following changes apply to the relationship of the odds:

$$\frac{\mathrm{P}(y_i = 1|x_{i1} + 1, \ldots)}{\mathrm{P}(y_i = 0|x_{i1} + 1, \ldots)} \Big/ \frac{\mathrm{P}(y_i = 1|x_{i1}, \ldots)}{\mathrm{P}(y_i = 0|x_{i1}, \ldots)} = \exp(\beta_1).$$

- $\beta_1 > 0$: $\mathrm{P}(y_i = 1)/\mathrm{P}(y_i = 0)$ increases,
- $\beta_1 < 0$: $\mathrm{P}(y_i = 1)/\mathrm{P}(y_i = 0)$ decreases,
- $\beta_1 = 0$: $\mathrm{P}(y_i = 1)/\mathrm{P}(y_i = 0)$ remains unchanged.

Deviance and deviance residuals

- The deviance is defined by

$$D = -2 \sum_{i=1}^{N} \{\ell_i(\hat{\mu}_i) - \ell_i(y_i)\}$$

  with $\hat{\mu}_i$ are the estimated expectations. $\ell_i(y_i)$ is the log-likelihood of the saturated model where the number of observations is equal to the number of parameters.

- The deviance residuals are defined as

$$d_i = \text{sign}(y_i - \hat{y}_i)\sqrt{-2(\ell_i(\hat{\mu}_i) - \ell_i(y_i))}.$$

Estimation in R

The function for fitting a binary regression model in R is:

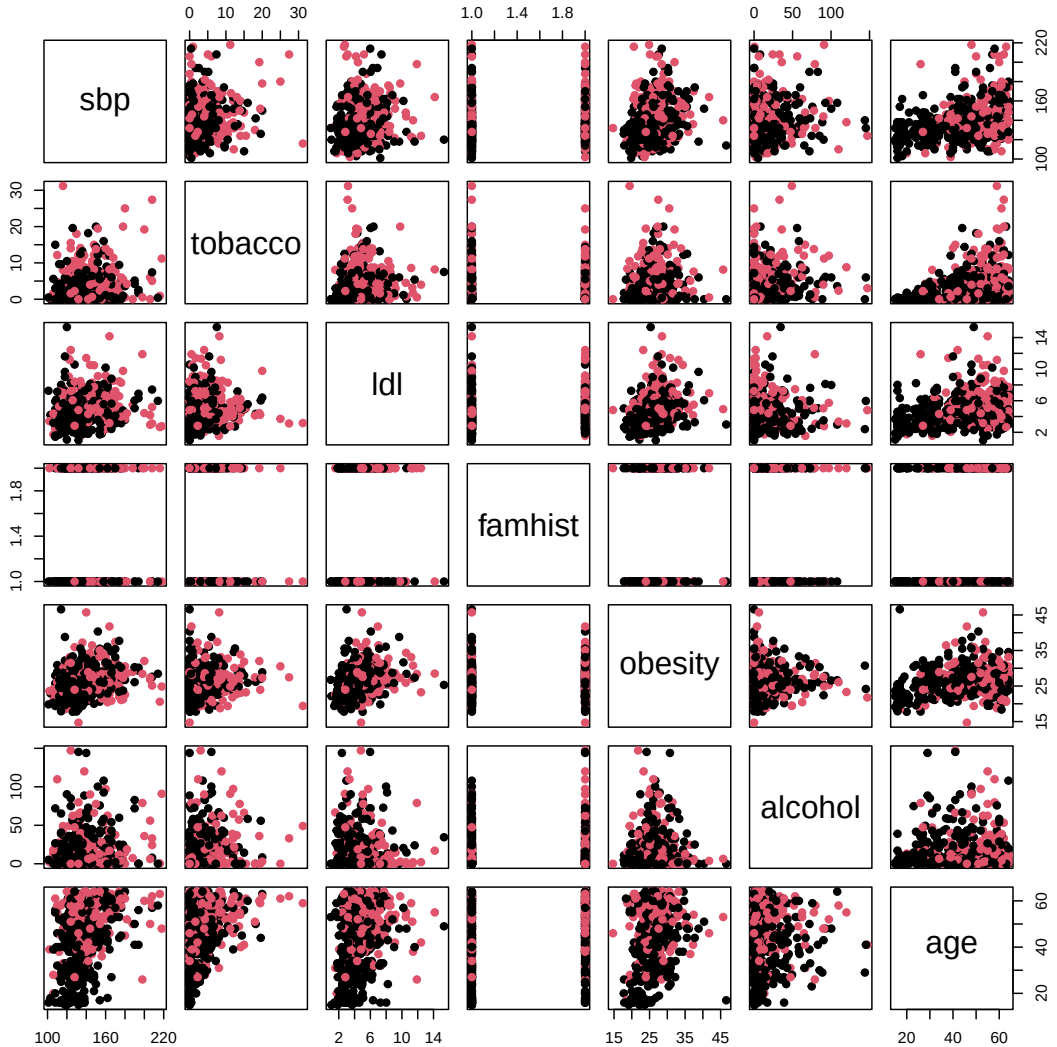
```
> glm(formula, family = binomial(link = "logit"), data,
+   weights, subset, na.action, ...)
```

- The formula is specified as for the linear model to model effects on the linear predictor.
- The link for the `binomial()` family can be specified to be logit, probit or complementary log-log.
- Maximum likelihood estimation is performed using an iterative procedure.

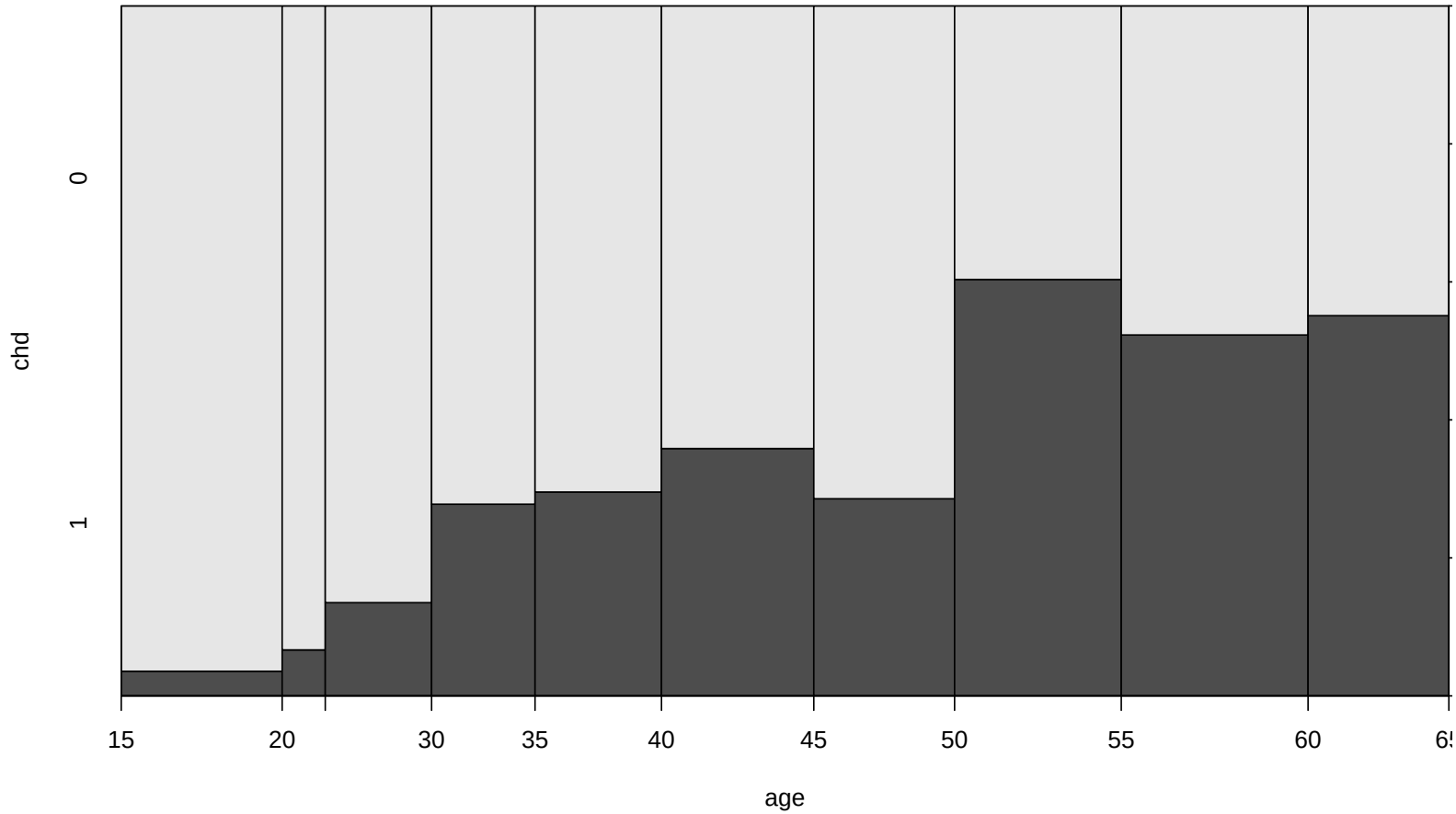
Example: South African Heart Disease

- Data set provided in the R package **ElemStatLearn**.
- The data set contains 462 observations.
- The dependent variable is `chd`, an indicator for coronary heart disease.
- The independent variables are the following risk factors:
  - systolic blood pressure (`sbp`);
  - cumulative tobacco (kg, `tobacco`);
  - low density lipoprotein cholesterol (`ldl`);
  - family history of heart disease (`famhist`);
  - obesity (`obesity`);
  - current alcohol consumption (`alcohol`);
  - age at onset (`age`).

Example: South African Heart Disease / 2



Example: South African Heart Disease / 3



Example: South African Heart Disease / 4

Full logistic regression model:

```
> model1 <- glm(chd ~ age, data = SAheart,
+   family = binomial())
```

Example: South African Heart Disease / 5

```
> summary(model1)

Call:
glm(formula = chd ~ age, family = binomial(), data = SAhear

Coefficients:
             Estimate Std. Error z value Pr(>|z|)
(Intercept) -3.521710   0.416031  -8.465  < 2e-16 ***
age          0.064108   0.008532   7.513 5.76e-14 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘

(Dispersion parameter for binomial family taken to be 1)

    Null deviance: 596.11  on 461  degrees of freedom
Residual deviance: 525.56  on 460  degrees of freedom
```

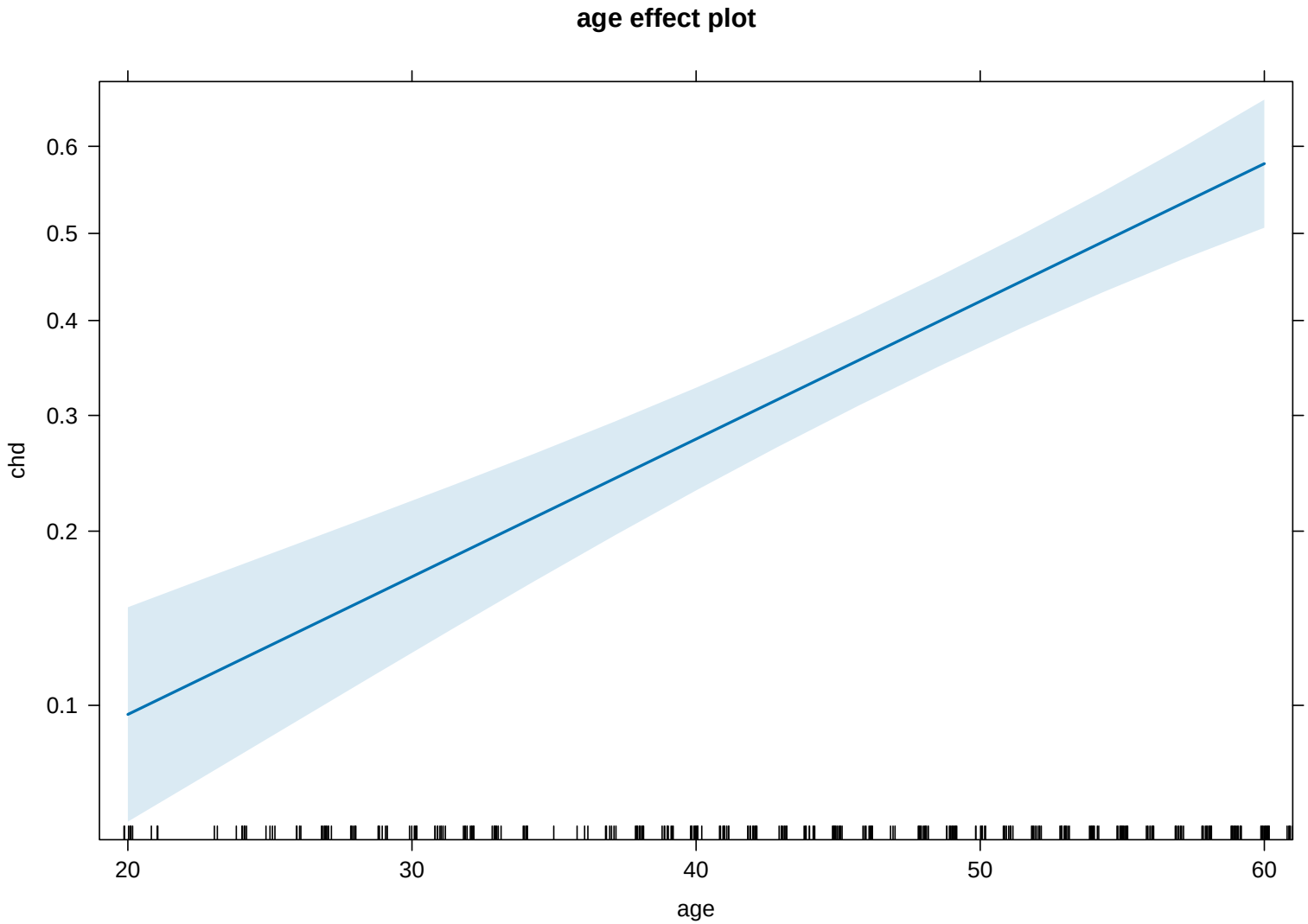
Example: South African Heart Disease / 6

```
AIC: 529.56

Number of Fisher Scoring iterations: 4
```

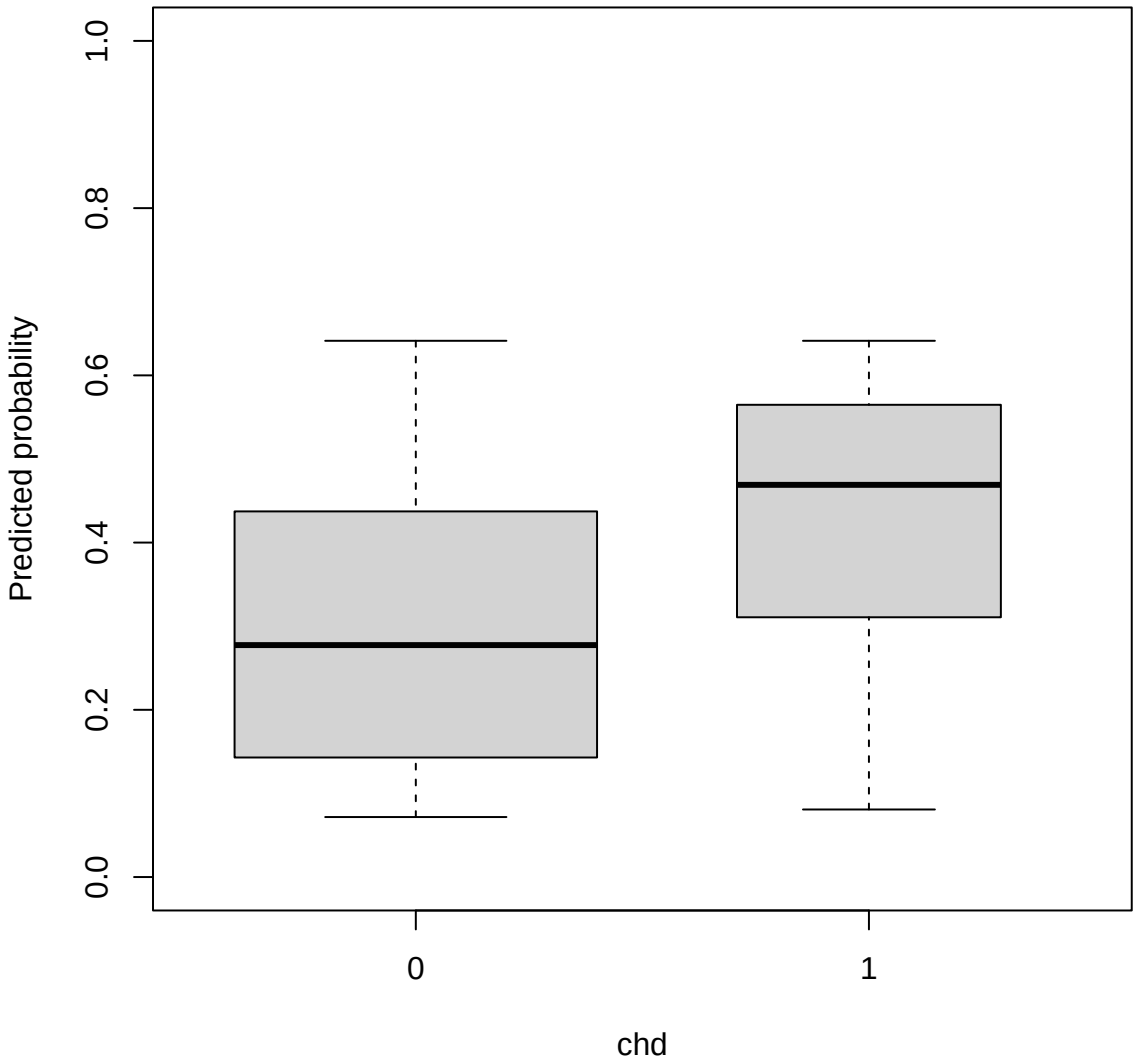
Example: South African Heart Disease / 7

```
> library("effects")
> Effect("age", model1)

 age effect
age
       20          30          40          50          60
0.09625471 0.16819565 0.27740126 0.42157561 0.58048669

> eff1 <- allEffects(model1)

> plot(eff1)
```

Example: South African Heart Disease / 8



Example: South African Heart Disease / 9



Example: South African Heart Disease / 10

```
> model2 <- glm(chd ~ age + famhist, data = SAheart,
+   family = binomial())
> model2

Call:  glm(formula = chd ~ age + famhist, family = binomial

Coefficients:
   (Intercept)             age  famhistPresent
       -3.7585          0.0597          0.9339

Degrees of Freedom: 461 Total (i.e. Null);  459 Residual
Null Deviance:              596.1
Residual Deviance: 506.7        AIC: 512.7
```

Example: South African Heart Disease / 11

```
> summary(model2)

Call:
glm(formula = chd ~ age + famhist, family = binomial(), dat

Coefficients:
                Estimate Std. Error z value Pr(>|z|)
(Intercept)    -3.758540   0.437062  -8.600  < 2e-16 ***
age             0.059705   0.008796   6.787 1.14e-11 ***
famhistPresent  0.933937   0.216312   4.318 1.58e-05 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘

(Dispersion parameter for binomial family taken to be 1)

    Null deviance: 596.11  on 461  degrees of freedom
```

Example: South African Heart Disease / 12

```
Residual deviance: 506.66  on 459  degrees of freedom
AIC: 512.66

Number of Fisher Scoring iterations: 4
```

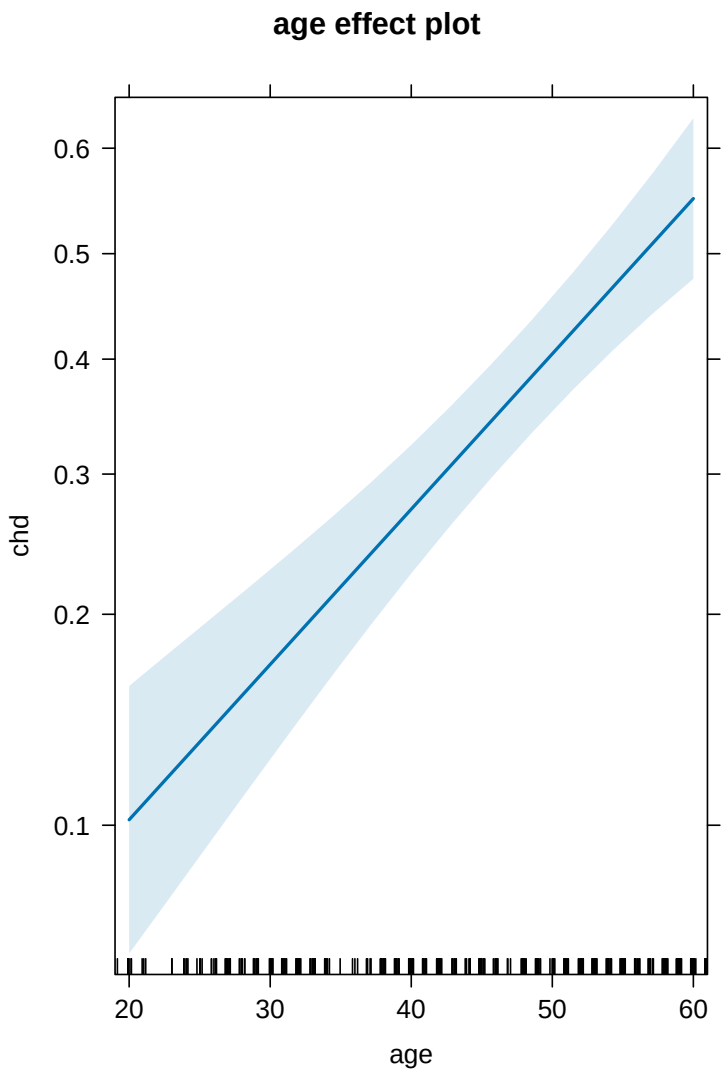
Example: South African Heart Disease / 13

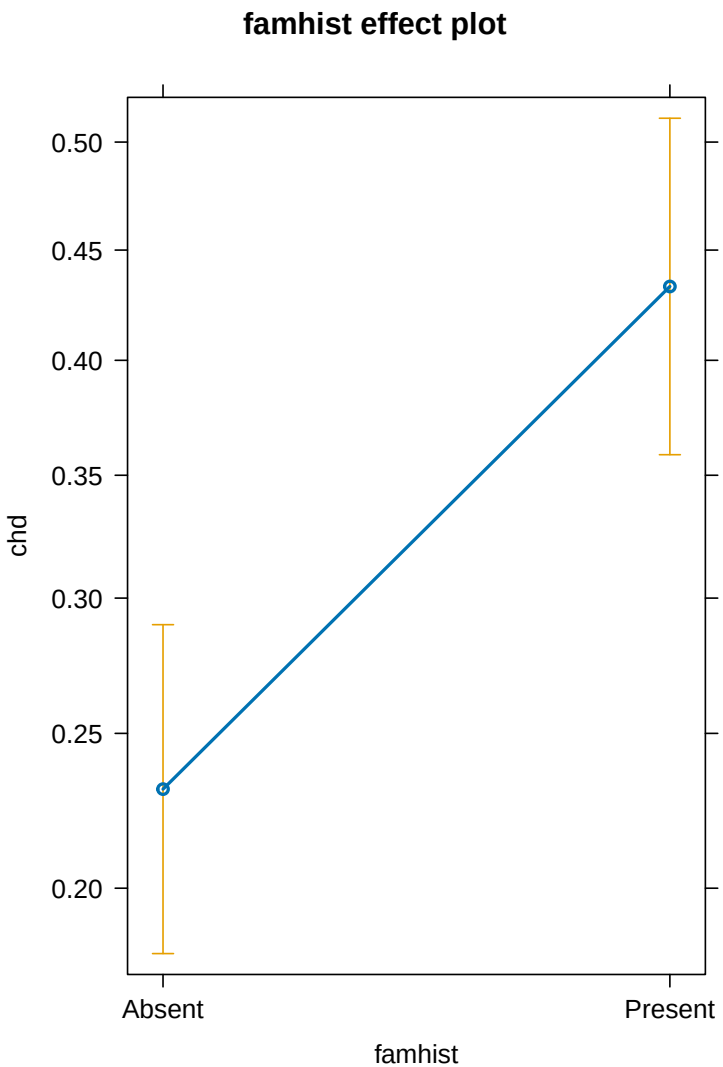
```
> anova(model1, model2)
Analysis of Deviance Table

Model 1: chd ~ age
Model 2: chd ~ age + famhist
  Resid. Df Resid. Dev Df Deviance  Pr(>Chi)
1       460     525.56
2       459     506.66  1   18.904 1.375e-05 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘
```

Example: South African Heart Disease / 14

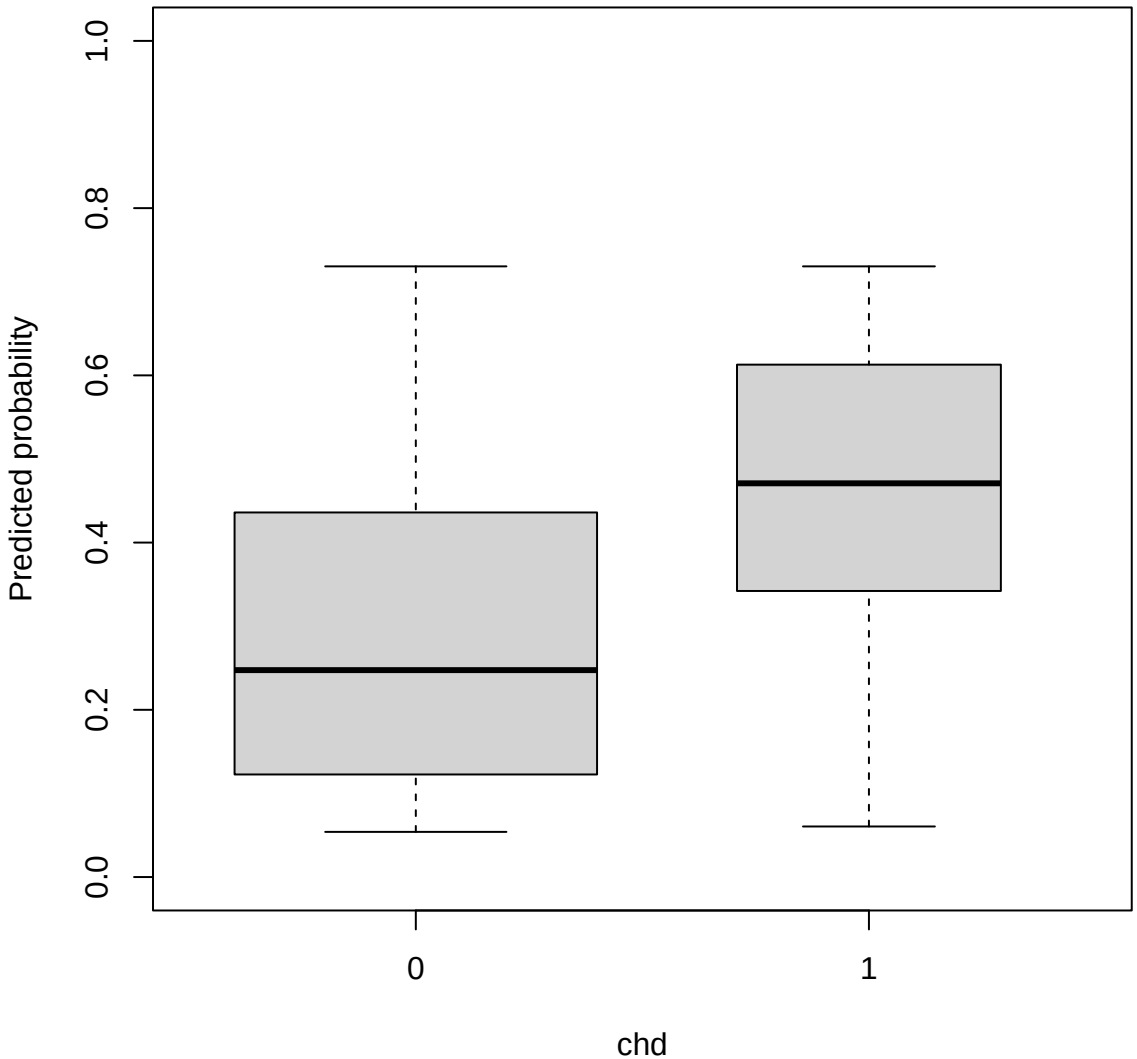
```
> eff2 <- allEffects(model2)
> eff2
 model: chd ~ age + famhist

 age effect
age
      20        30        40        50        60
0.1018973 0.1708984 0.2724500 0.4048778 0.5527688

 famhist effect
famhist
   Absent   Present
0.2310792 0.4333267
```

Example: South African Heart Disease / 15





Example: South African Heart Disease / 16



(Quasi-)complete separation

- Complete separation happens when the outcome variable separates a predictor variable or a combination of predictor variables completely.
- Quasi-complete separation happens when the outcome variable separates a predictor variable or a combination of predictor variables to a certain degree.
- (Quasi-)complete separation leads to large coefficient estimates and standard errors.
- In R no check for (quasi-)complete separation is performed by default. In general issues are indicated by a warning about fitted probabilities being numerically 0 or 1.
- Strategies to deal with (quasi-)complete separation:
  - Assess if the outcome variable is not a dichotomous version of a variable in the model.
  - One might decide not to include the problematic variable in the model. But this may lead to biased estimates.

Example: South African Heart Disease

```
> SAheart0 <- subset(SAheart,
+   (age <= 50 & chd == 0) | (age >= 50 & chd == 1))
> model0 <- glm(chd ~ famhist + age,
+   data = SAheart0, family = binomial())
Warning message:
glm.fit: fitted probabilities numerically 0 or 1 occurred
```

Example: South African Heart Disease / 2

```
> summary(model0)

Call:
glm(formula = chd ~ famhist + age, family = binomial(), dat

Coefficients:
                 Estimate Std. Error z value Pr(>|z|)
(Intercept)    -9.526e+02  1.124e+05  -0.008    0.993
famhistPresent  2.877e-01  1.528e+00   0.188    0.851
age             1.904e+01  2.249e+03   0.008    0.993

(Dispersion parameter for binomial family taken to be 1)

    Null deviance: 399.571  on 324  degrees of freedom
Residual deviance:  10.549  on 322  degrees of freedom
AIC: 16.549
```

Example: South African Heart Disease / 3

```
Number of Fisher Scoring iterations: 25
```

Generalized linear regression models

General model definition

The linear model and the regression models for non-normal response variables have common properties that can be summarized in a unified framework:

1. The mean $\mu = \mathrm{E}(y)$ of the response y is connected with the linear predictor $\eta = \boldsymbol{x}^\top \boldsymbol{\beta}$ by a response function h or by a link function $g = h^{-1}$:

$$\mu = h(\eta) \qquad \text{or} \qquad \eta = g(\mu).$$

2. The distribution of the response variables (normal, binomial, Poisson, and gamma distribution) can be written in the form of a **univariate exponential family**.

Univariate exponential families

- The density of a univariate exponential dispersion family for the response variable y is defined by

$$f(y|\theta) = \exp\left(\frac{y\theta - b(\theta)}{a(\phi)} + c(y, \phi)\right).$$

- The parameter θ is called the natural or canonical parameter.
- The second parameter ϕ is a dispersion parameter.
- For the function $b(\theta)$ it is required that $f(y|\theta)$ can be normalized and the first and second derivative $b'(\theta)$ and $b''(\theta)$ exist.
- It can be shown that

$$\mathrm{E}(y) = \mu = b'(\theta), \qquad \mathrm{VAR}(y) = a(\phi)b''(\theta).$$

Univariate exponential families / 2

Examples:

Distribution		$\theta(\mu)$	$b(\theta)$	$a(\phi)$
Normal	$N(\mu, \sigma^2)$	μ	$\theta^2/2$	σ^2
Bernoulli	$B(1, \pi)$	$\log(\pi/(1-\pi))$	$\log(1+\exp(\theta))$	1
Poisson	$Po(\lambda)$	$\log(\lambda)$	$\exp(\theta)$	1

Generalized linear model

Distributional assumptions

For given covariates $\boldsymbol{x}_i = (1, x_{i1}, \ldots, x_{ik})^\top$, the response variables are (conditionally) independent and the (conditional) density of y_i belongs to the exponential family with

$$f(y_i|\theta_i) = \exp\left(\frac{y_i\theta_i - b(\theta_i)}{a(\phi)} + c(y_i, \phi)\right).$$

The parameter θ_i is called the natural parameter and ϕ the dispersion parameter. For $\mathrm{E}(y_i) = \mu_i$ and $\mathrm{VAR}(y_i)$, we have

$$\mathrm{E}(y_i) = \mu_i = b'(\theta_i), \qquad \mathrm{VAR}(y_i) = a(\phi)b''(\theta_i).$$

Generalized linear model / 2

Structural assumptions

The (conditional) mean μ_i is connected to the linear predictor $\eta_i = \boldsymbol{x}_i^\top \boldsymbol{\beta} = \beta_0 + \beta_1 x_{i1} + \ldots + \beta_k x_{ik}$ through

$$\mu_i = h(\eta_i) = h(\boldsymbol{x}_i^\top \boldsymbol{\beta}), \qquad \text{or} \qquad \eta_i = g(\mu_i),$$

where

h is a (one-to-one and twice differentiable) response function and

g is the link function, i.e., the inverse $g = h^{-1}$.

Maximum likelihood estimation

The ML estimator $\hat{\boldsymbol{\beta}}$ maximizes the (log-)likelihood and is defined as the solution

$$\boldsymbol{s}(\hat{\boldsymbol{\beta}}) = 0$$

of the score function given by

$$\boldsymbol{s}(\boldsymbol{\beta}) = \sum_{i=1}^{N} \boldsymbol{x}_i \frac{h'(\eta_i)}{\text{VAR}(y_i)}(y_i - \mu_i) = \boldsymbol{X}^\top \boldsymbol{D}\boldsymbol{\Sigma}^{-1}(\boldsymbol{y} - \boldsymbol{\mu}),$$

where

$$\boldsymbol{D} = \text{diag}(h'(\eta_1), \ldots, h'(\eta_N)), \quad \boldsymbol{\Sigma} = \text{diag}(\text{VAR}(y_1), \ldots, \text{VAR}(y_N))$$
$$\boldsymbol{\mu} = (\mu_1, \ldots, \mu_N)^\top.$$

Maximum likelihood estimation / 2

The Fisher matrix is

$$\boldsymbol{F}(\boldsymbol{\beta}) = \sum_{i=1}^{N} \boldsymbol{x}_i \boldsymbol{x}_i^\top \tilde{w}_i = \boldsymbol{X}^\top \boldsymbol{W} \boldsymbol{X},$$

where

$$\boldsymbol{W} = \text{diag}(\tilde{w}_1, \ldots, \tilde{w}_N)$$

is the diagonal matrix of working weights

$$\tilde{w}_i = \frac{(h'(\eta_i))^2}{\text{VAR}(y_i)}.$$

Maximum likelihood estimation / 3

The ML estimator $\hat{\boldsymbol{\beta}}$ is obtained iteratively using Fisher scoring in form of iteratively weighted least squares estimates

$$\hat{\boldsymbol{\beta}}^{(t+1)} = (\boldsymbol{X}^\top \boldsymbol{W}^{(t)} \boldsymbol{X})^{-1} \boldsymbol{X}^\top \boldsymbol{W}^{(t)} \tilde{\boldsymbol{y}}^{(t)}, \quad t = 0, 1, 2, \ldots,$$

with working observations given by

$$\tilde{y}_i^{(t)} = \hat{\eta}_i^{(t)} + \frac{(y_i - h(\hat{\eta}_i^{(t)}))}{h'(\hat{\eta}_i^{(t)})}.$$

Likelihood inference

Asymptotic properties of the ML estimator

Let $\hat{\boldsymbol{\beta}}_N$ denote the ML estimator based on a sample of size N. Under regularity conditions, $\hat{\boldsymbol{\beta}}_N$ is consistent and asymptotically normal:

$$\hat{\boldsymbol{\beta}}_N \overset{a}{\sim} \mathrm{N}(\boldsymbol{\beta}, \boldsymbol{F}^{-1}(\boldsymbol{\beta})).$$

This result holds even if the estimator $\boldsymbol{F}(\hat{\boldsymbol{\beta}})$ replaces $\boldsymbol{F}(\boldsymbol{\beta})$.

Regularized generalized linear models

Regularized GLMs

- Regularized OLS estimation of linear models corresponds to regularized ML estimation.
- The regularized ML criterion can directly be extended to the GLM case.
- The regularized estimator is obtained by solving:

$$\hat{\beta}^{\text{pen}} = \arg\min_{\beta} \left\{ -\frac{1}{N} \ell(\beta, \phi | \boldsymbol{y}, \boldsymbol{X}) + \lambda P(\beta) \right\},$$

  where
  - twice the negative log-likelihood is used instead of the residual sum of squares;
  - the penalty function $P()$ penalizes the "length" of the regression coefficient vector β, e.g., represents the ridge, lasso or elastic net penalty.

Regularized GLMs / 2

- GLMs are usually fitted using iterative weighted least squares (IWLS).
- Estimation of regularized GLMs based on a pathwise coordinate descent algorithm:
  - Compute the solution for a decreasing sequence of penalty parameter values.
  - Initialize coordinate descent algorithm with previous solution.

Software in R

Package **glmnet**:

- Fits a generalized linear model via penalized maximum likelihood.
- The regularization path is computed for the lasso, ridge or elastic net penalty at a grid of values for the regularization parameter λ.
- Implements pathwise coordinate optimization.
- No formula interface is provided.
- By default an intercept is added and the covariates are standardized.

Example: South African Heart Disease

Full logistic regression model:

```
> full.model <- glm(chd ~ ., data = SAheart,
+   family = binomial())
> printCoefmat(round(coef(summary(full.model)),
+   digits = 2))
                  Estimate Std. Error z value Pr(>|z|)
(Intercept)          -4.13       0.96   -4.28   <2e-16 ***
sbp                   0.01       0.01    1.02     0.31
tobacco               0.08       0.03    3.03   <2e-16 ***
ldl                   0.18       0.06    3.22   <2e-16 ***
famhistPresent        0.94       0.22    4.18   <2e-16 ***
obesity              -0.03       0.03   -1.19     0.24
alcohol               0.00       0.00    0.14     0.89
age                   0.04       0.01    4.18   <2e-16 ***
```

Example: South African Heart Disease / 2

```
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘
```

Stepwise selected logistic regression model:

```
> step.model <- step(full.model, trace = 0)
> printCoefmat(round(coef(summary(step.model)),
+   digits = 2))
                  Estimate Std. Error z value  Pr(>|z|)
(Intercept)          -4.20       0.50   -8.44 < 2.2e-16 ***
tobacco               0.08       0.03    3.16 < 2.2e-16 ***
ldl                   0.17       0.05    3.09 < 2.2e-16 ***
famhistPresent        0.92       0.22    4.14 < 2.2e-16 ***
age                   0.04       0.01    4.52 < 2.2e-16 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘
```

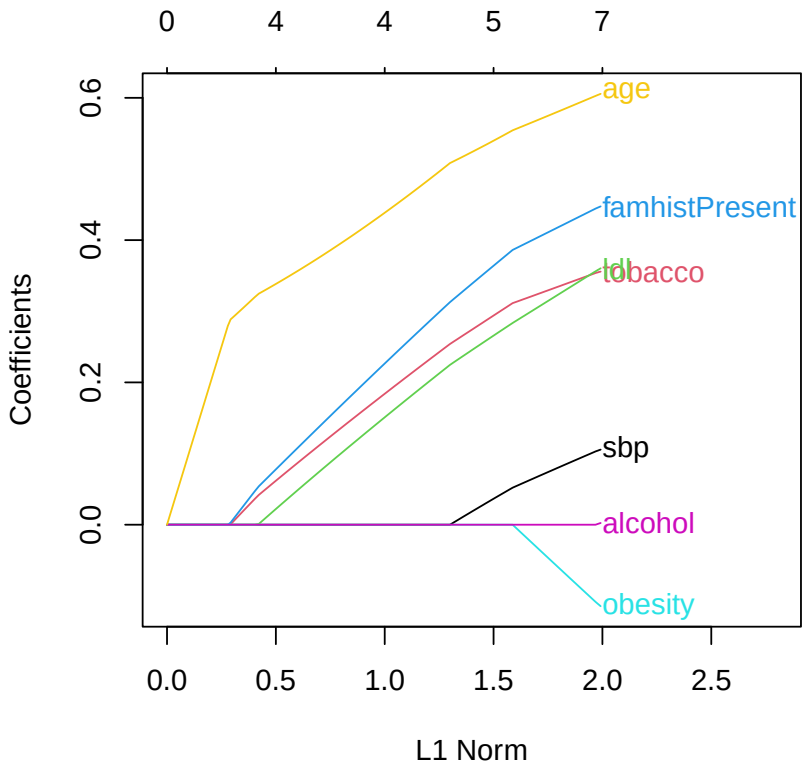
Example: South African Heart Disease / 3

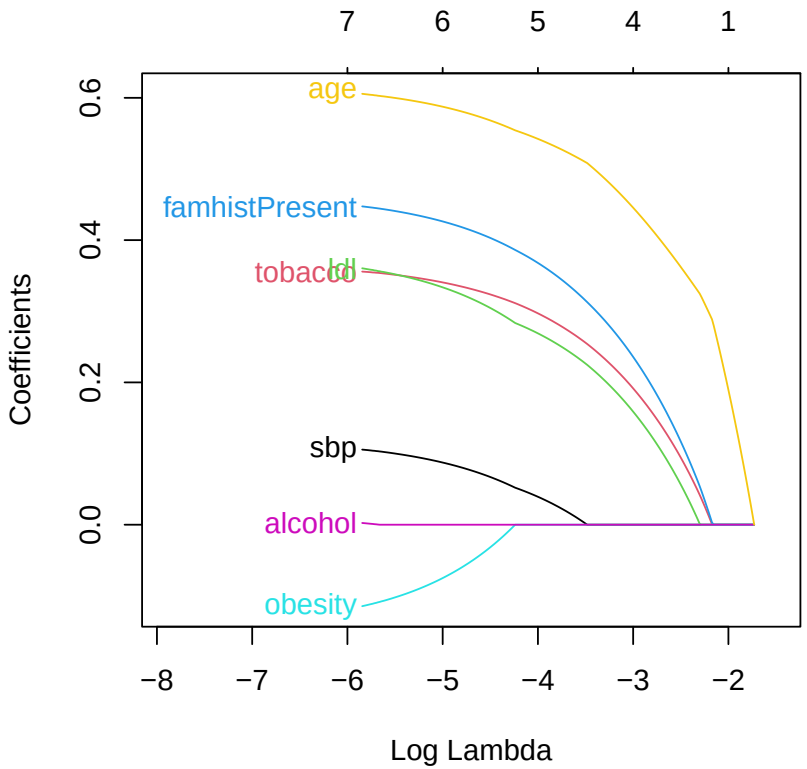
```
> mf <- model.frame(chd ~ ., data = SAheart)
> X <- model.matrix(chd ~ ., data = mf)[, -1]
> X <- scale(X)
> y <- model.response(mf)
```

Lasso regression:

```
> library("glmnet")
> model <- glmnet(X, y, family = "binomial",
+   nlambda = 500)
```

Example: South African Heart Disease / 4





Example: South African Heart Disease / 5

Ridge regression:

```
> model <- glmnet(X, y, family = "binomial",
+   alpha = 0, nlambda = 500)
```