Generalized linear models

Linear regression:

- $\mathbf{y} = \mathbf{X}\theta + \epsilon$, with $\epsilon \sim \mathcal{N}(0, \sigma^2)$
- $E[\mathbf{y}] = \mu = \mathbf{X}\theta$

This is appropriate when $\mathbf{y} \sim \mathcal{N}(\mu, \sigma^2)$, e.g. $\mathbf{y}$ is continuous, can be both positive and negative, etc.

- In glm, $\mathbf{y}$ is assumed to be generated from a particular distribution of the exponential family.
- $E[\mathbf{y}] = \mu = g^{-1}(\mathbf{X}\theta)$
- g is called **link function**

$E[\mathbf{y}] = \mu = g^{-1}(\mathbf{X}\theta)$

Link functions:

- identity link: $g(\mu) = \mu = \mathbf{X}\theta$
- logit link: $g(\mu) = log\left(\frac{\mu}{1-\mu}\right) = \mathbf{X}\theta \rightarrow \mu = \frac{e^{\mathbf{X}\theta}}{1+e^{\mathbf{X}\theta}}$

Logit link is used in logistic regression.

- probit link: $g(\mu) = \Phi^{-1}(\mu) = \mathbf{X}\theta \rightarrow \mu = \Phi(\mathbf{X}\theta)$

Φ is the cumulative function of the standard normal distribution.

- log link: $g(\mu) = log(\mu) = \mathbf{X}\theta \rightarrow \mu = e^{\mathbf{X}\theta}$

Log link is used in Poisson regression.

glm in R

In glm you can specify the family of y and the link function (see help(glm) and help(family)).

E.g.

- Linear regression: family=gaussian
- Binary response (logistic regession): family=binomial
- Counts: family=poisson

Simulation example

In [2]:

```
set.seed(600)

p53_expression <- rnorm(100)

p53_theta <- 1 + 3 * p53_expression  # linear model

pr <- exp( p53_theta ) / ( 1 + exp( p53_theta ) )  # inverse logit function

Healthy_Disease <- rbinom(n=length( p53_expression ), size=1, prob=pr) # bernou
lli response variable

data_sim <- data.frame(Healthy_Disease = Healthy_Disease, p53_expression = p53_
expression)
```

In [3]:

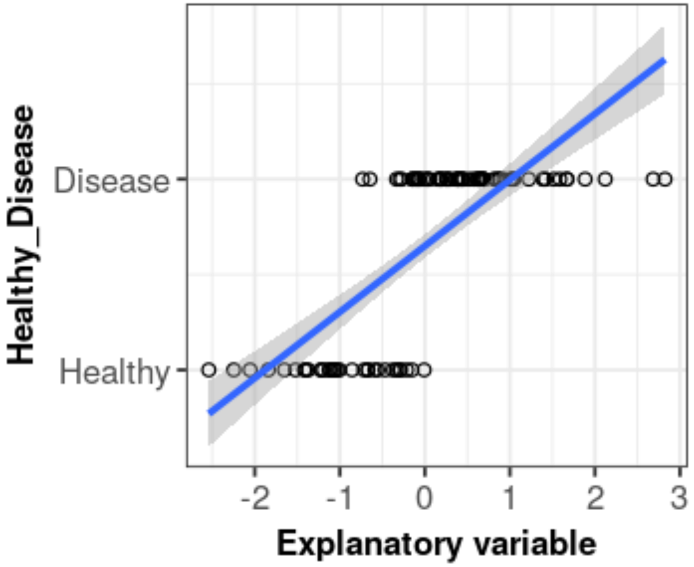
```
head(data_sim)
```

A data.frame: 6 × 2

Healthy_Disease	p53_expression
<int>	<dbl>
0	-1.12014151
1	0.19827413
1	0.64516581
0	-0.15145186
1	-0.01841739
1	0.01770806

In [4]:

```
library(ggplot2)

ggplot(data_sim, aes(x=p53_expression, y=Healthy_Disease)) +

    geom_point(shape=1) +

    geom_smooth(method = "glm", method.args = list(family = "gaussian"), se=TRUE) +

    scale_y_continuous( breaks=c(0,1), labels=c("Healthy","Disease") ) +

    xlab( "Explanatory variable" ) + theme_bw()+

    theme( axis.text = element_text(size=10),
           axis.title = element_text(size=10, face="bold"))
```



In [5]:

```
model_lm <- lm( Healthy_Disease ~ p53_expression, data = data_sim )

summary( model_lm )
```

```
Call:
lm(formula = Healthy_Disease ~ p53_expression, data = data_sim)

Residuals:
     Min       1Q   Median       3Q      Max 
-0.64712 -0.24359  0.06488  0.24895  0.60343 

Coefficients:
               Estimate Std. Error t value Pr(>|t|)    
(Intercept)     0.65010    0.03184   20.42   <2e-16 ***
p53_expression  0.34623    0.03150   10.99   <2e-16 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 0.3179 on 98 degrees of freedom
Multiple R-squared:  0.5521,	Adjusted R-squared:  0.5475 
F-statistic: 120.8 on 1 and 98 DF,  p-value: < 2.2e-16
```

In [6]:

```
model_glm <- glm( Healthy_Disease ~ p53_expression, data = data_sim, family = "
gaussian" )

summary( model_glm )
```

```
Call:
glm(formula = Healthy_Disease ~ p53_expression, family = "gaussian",
    data = data_sim)

Deviance Residuals:
     Min        1Q    Median        3Q       Max
-0.64712  -0.24359   0.06488   0.24895   0.60343

Coefficients:
               Estimate Std. Error t value Pr(>|t|)
(Intercept)     0.65010    0.03184   20.42   <2e-16 ***
p53_expression  0.34623    0.03150   10.99   <2e-16 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

(Dispersion parameter for gaussian family taken to be 0.1010477)

    Null deviance: 22.1100  on 99  degrees of freedom
Residual deviance:  9.9027  on 98  degrees of freedom
AIC: 58.551

Number of Fisher Scoring iterations: 2
```

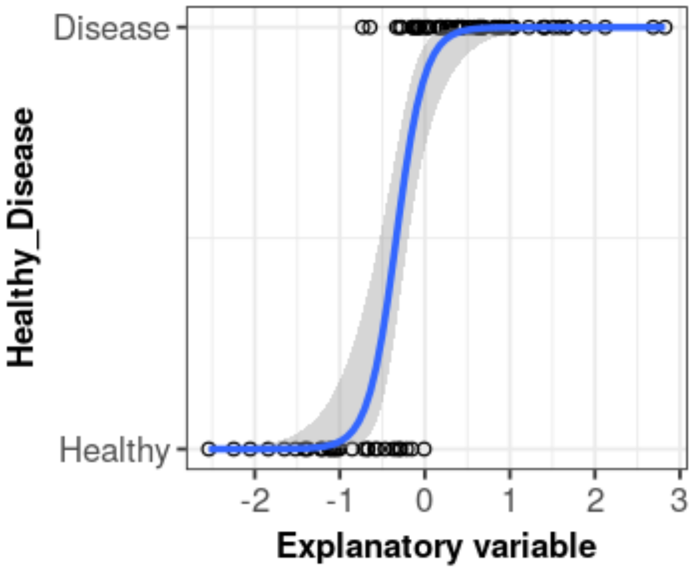
In [7]:

```
model_glm <- glm( Healthy_Disease ~ p53_expression, data = data_sim, family = g
aussian( link = identity ) )

summary( model_glm  )
```

```
Call:
glm(formula = Healthy_Disease ~ p53_expression, family = gaussian(link = ident
ity), 
    data = data_sim)

Deviance Residuals: 
     Min        1Q    Median        3Q       Max  
-0.64712  -0.24359   0.06488   0.24895   0.60343  

Coefficients:
               Estimate Std. Error t value Pr(>|t|)    
(Intercept)     0.65010    0.03184   20.42   <2e-16 ***
p53_expression  0.34623    0.03150   10.99   <2e-16 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

(Dispersion parameter for gaussian family taken to be 0.1010477)

    Null deviance: 22.1100  on 99  degrees of freedom
Residual deviance:  9.9027  on 98  degrees of freedom
AIC: 58.551

Number of Fisher Scoring iterations: 2
```

In [8]:

```
ggplot(data_sim, aes(x=p53_expression, y=Healthy_Disease)) +

    geom_point(shape=1) +

    geom_smooth(method="glm", method.args=list(family = "binomial"), se=TRUE)+

    scale_y_continuous(breaks=c(0,1), labels=c("Healthy","Disease"))+

    xlab("Explanatory variable") + theme_bw() +

    theme(axis.text=element_text(size=10),
            axis.title=element_text(size=10,face="bold"))
```



In [9]:

```
model_glm <- glm(Healthy_Disease ~ p53_expression, data = data_sim, family = "binomial")

summary(model_glm)
```

```
Call:
glm(formula = Healthy_Disease ~ p53_expression, family = "binomial", 
    data = data_sim)

Deviance Residuals: 
     Min        1Q    Median        3Q       Max  
-2.05774  -0.05788   0.03545   0.20972   2.22866  

Coefficients:
               Estimate Std. Error z value Pr(>|z|)    
(Intercept)      2.0410     0.5752   3.549 0.000387 ***
p53_expression   6.0597     1.5224   3.980 6.88e-05 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

(Dispersion parameter for binomial family taken to be 1)

    Null deviance: 126.836  on 99  degrees of freedom
Residual deviance:  35.737  on 98  degrees of freedom
AIC: 39.737

Number of Fisher Scoring iterations: 8
```

In [ ]: