

MSE 598-DM - Statistical Modeling

# Generalized Linear Models

Dr. Bo Li

Department of Statistics

University of Illinois at Urbana-Champaign

Spring 2022

Slides credit to Dr. Trevor Park, Dr. Jeff Douglass and Dr. Steve Culpepper

## A linear model

$$Y = \underbrace{\alpha + \beta_1 x_1 + \cdots + \beta_p x_p}_{E(Y) = \mu} + \varepsilon$$

is usually **not** appropriate if  $Y$  is binary or a count.

To go further, need regression-type models for binary or count-type responses.

Assumed: Some exposure to linear regression and linear algebra.

# The Generalized Linear Model (GLM)

Seek to model independent observations

$$Y_1, \dots, Y_n$$

of a **response variable**, in terms of corresponding vectors

$$\mathbf{x}_i = (x_{i1}, \dots, x_{ip}) \quad i = 1, \dots, n$$

of values of  $p$  **explanatory variables**.

(All variables are represented by numbers, possibly some are dummy variables.)

- ▶ Random component: density of  $Y_i$  from a **natural exponential family**

$$f(y_i; \theta_i) = a(\theta_i) b(y_i) \exp(y_i Q(\theta_i))$$

$Q(\theta_i)$  is the **natural parameter**.

- ▶ Random component: density of  $Y_i$  from a **natural exponential family**

$$f(y_i; \theta_i) = a(\theta_i) b(y_i) \exp(y_i Q(\theta_i))$$

$Q(\theta_i)$  is the **natural parameter**.

- ▶ Systematic component: the **linear predictor**

$$\eta_i = \alpha + \beta_1 x_{i1} + \cdots + \beta_p x_{ip}$$

with parameters  $\alpha, \beta_1, \dots, \beta_p$  (**coefficients**)

$Y_i$  will depend on  $\mathbf{x}_i$  only through  $\eta_i$

- **Link function:** monotonic, differentiable  $g$  such that

$$g(\mu_i) = \eta_i \quad \text{where} \quad \mu_i = E(Y_i)$$

Note: Ordinary linear models use the **identity link**:

$$g(\mu) = \mu$$

The **canonical link** satisfies

$$g(\mu_i) = Q(\theta_i)$$

which means

$$Q(\theta_i) = \alpha + \beta_1 x_{i1} + \cdots + \beta_p x_{ip}$$

## Binary Regression

$$Y_i \sim \text{Bernoulli}(\pi_i) \quad (\theta_i = \pi_i)$$

$$f(y_i; \pi_i) = \begin{cases} 1 - \pi_i & y_i = 0 \\ \pi_i & y_i = 1 \end{cases}$$

## Binary Regression

$$Y_i \sim \text{Bernoulli}(\pi_i) \quad (\theta_i = \pi_i)$$

$$\begin{aligned} f(y_i; \pi_i) &= \begin{cases} 1 - \pi_i & y_i = 0 \\ \pi_i & y_i = 1 \end{cases} \\ &= \pi_i^{y_i} (1 - \pi_i)^{1-y_i} \end{aligned}$$



## Binary Regression

$$Y_i \sim \text{Bernoulli}(\pi_i) \quad (\theta_i = \pi_i)$$

$$\begin{aligned} f(y_i; \pi_i) &= \begin{cases} 1 - \pi_i & y_i = 0 \\ \pi_i & y_i = 1 \end{cases} \\ &= \pi_i^{y_i} (1 - \pi_i)^{1-y_i} \\ &= (1 - \pi_i) \left( \frac{\pi_i}{1 - \pi_i} \right)^{y_i} \end{aligned}$$

## Binary Regression

$$Y_i \sim \text{Bernoulli}(\pi_i) \quad (\theta_i = \pi_i)$$

$$f(y_i; \pi_i) = \begin{cases} 1 - \pi_i & y_i = 0 \\ \pi_i & y_i = 1 \end{cases}$$

$$= \pi_i^{y_i} (1 - \pi_i)^{1-y_i}$$

$$= (1 - \pi_i) \left( \frac{\pi_i}{1 - \pi_i} \right)^{y_i}$$

$$= (1 - \pi_i) \exp \left( y_i \ln \left( \frac{\pi_i}{1 - \pi_i} \right) \right)$$

So

$$a(\pi) = 1 - \pi \qquad b(y) = 1$$

$$Q(\pi) = \ln\left(\frac{\pi}{1 - \pi}\right) = \text{logit}(\pi)$$

The natural parameter is the **log odds**.

Note:

$$\mu_i = E(Y_i) = \pi_i$$

Canonical link:

$$g(\pi) = Q(\pi) = \text{logit}(\pi)$$

which leads to **logistic regression**.

## Poisson Regression

$$Y_i \sim \text{Poisson}(\mu_i) \quad (\theta_i = \mu_i)$$

Note:

$$\mu_i = \text{E}(Y_i)$$

$$\begin{aligned} f(y_i; \mu_i) &= \frac{\mu_i^{y_i}}{y_i!} e^{-\mu_i} \\ &= e^{-\mu_i} \frac{1}{y_i!} \exp(y_i \ln \mu_i) \end{aligned}$$

So

$$a(\mu) = e^{-\mu} \qquad b(y) = \frac{1}{y!}$$

$$Q(\mu) = \ln \mu$$

The natural parameter is the log-mean.

Canonical link:

$$g(\mu) = Q(\mu) = \ln \mu$$

which gives the **(Poisson) loglinear model**.

## Fitting GLMs

Usually by **maximum likelihood**: find

$$\hat{\alpha}, \hat{\beta}_1, \dots, \hat{\beta}_p$$

maximizing

$$\prod_{i=1}^n f(y_i; \theta_i)$$

Explicit solutions exist only in special cases, so need numerical methods (Agresti, Sec. 4.6):

- ▶ Newton-Raphson
- ▶ Fisher Scoring

Main R function: `glm()`

# Binary Response

$$Y_i \sim \text{Bernoulli}(\pi(\mathbf{x}_i))$$

$$\mathbb{E}(Y_i) = \pi(\mathbf{x}_i) \quad \text{var}(Y_i) = \pi(\mathbf{x}_i)(1 - \pi(\mathbf{x}_i))$$

► Identity link:

$$\pi(\mathbf{x}_i) = \alpha + \beta_1 x_{i1} + \cdots + \beta_p x_{ip}$$

# Binary Response

$$Y_i \sim \text{Bernoulli}(\pi(\mathbf{x}_i))$$

$$\mathbb{E}(Y_i) = \pi(\mathbf{x}_i) \quad \text{var}(Y_i) = \pi(\mathbf{x}_i)(1 - \pi(\mathbf{x}_i))$$

► Identity link:

$$\pi(\mathbf{x}_i) = \alpha + \beta_1 x_{i1} + \cdots + \beta_p x_{ip}$$

► Log link:

$$\ln(\pi(\mathbf{x}_i)) = \alpha + \beta_1 x_{i1} + \cdots + \beta_p x_{ip}$$



# Binary Response

$$Y_i \sim \text{Bernoulli}(\pi(\mathbf{x}_i))$$

$$E(Y_i) = \pi(\mathbf{x}_i) \quad \text{var}(Y_i) = \pi(\mathbf{x}_i)(1 - \pi(\mathbf{x}_i))$$

- Identity link:

$$\pi(\mathbf{x}_i) = \alpha + \beta_1 x_{i1} + \cdots + \beta_p x_{ip}$$

- Log link:

$$\ln(\pi(\mathbf{x}_i)) = \alpha + \beta_1 x_{i1} + \cdots + \beta_p x_{ip}$$

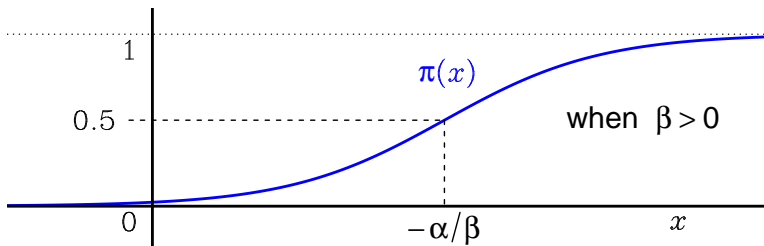
- Canonical link: (logistic regression)

$$\text{logit}(\pi(\mathbf{x}_i)) = \alpha + \beta_1 x_{i1} + \cdots + \beta_p x_{ip}$$

When  $p = 1$ ,

$$\text{logit}(\pi(x)) = \alpha + \beta x \Leftrightarrow \text{odds}(\pi(x)) = e^{\alpha + \beta x}$$

$$\Leftrightarrow \pi(x) = \frac{e^{\alpha + \beta x}}{1 + e^{\alpha + \beta x}}$$



## R Example: Psych Data

$$Y_i = \begin{cases} 0 & \text{if subject } i \text{ healthy (30)} \\ 1 & \text{if subject } i \text{ mentally ill (15)} \end{cases}$$

Explanatory variables

$$x_1, x_2, x_3, x_4, x_5$$

are scores (1 to 4) on five questions, where higher values are expected to be healthier.

```
> psych <- read.table("psych.txt", header=TRUE)
```

```
> head(psych)
```

	ill	x1	x2	x3	x4	x5
1	1	2	2	2	2	2
2	1	2	2	2	1	2
3	1	1	1	2	1	1
4	1	2	2	2	1	2
5	1	1	1	2	1	2
6	1	1	1	2	1	1

We will use `glm(family=binomial)`, which uses the canonical (logit) link function by default ...

```
> psychfit1 <- glm(ill ~ x1 + x2 + x3 + x4 + x5, family=binomial, data=psych)
```

```
> summary(psychfit1)
```

...

Coefficients:

	Estimate	Std. Error	z value	Pr(> z )
(Intercept)	15.2226	6.3321	2.404	0.0162 *
x1	-0.6757	0.8045	-0.840	0.4009
x2	-1.1471	1.0622	-1.080	0.2802
x3	-2.9016	2.2733	-1.276	0.2018
x4	-1.0316	0.7080	-1.457	0.1451
x5	-2.0280	1.2023	-1.687	0.0917 .

---

Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1

...

Number of Fisher Scoring iterations: 7

Q: What are  $\hat{\alpha}, \hat{\beta}_1, \dots, \hat{\beta}_5$ ?

All effects have the same direction, but none appears significant. (Possible collinearity?)

Try using the total score only:

```
> xsum <- apply(psych[,2:6], 1, sum)
> psychfit2 <- glm(ill ~ xsum, family=binomial, data=psych)
> summary(psychfit2)

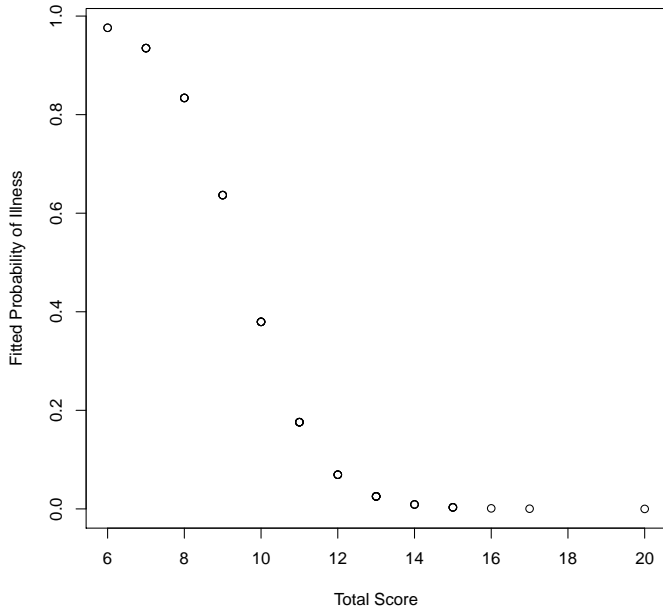
...

Coefficients:
              Estimate Std. Error z value Pr(>|z|)
(Intercept)  10.0331      3.1448   3.190 0.001421 **
xsum         -1.0524      0.3177  -3.313 0.000924 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

...
```

Plot fitted probabilities of illness versus the total score:

```
> plot(xsum, fitted(psychfit2), xlab="Total Score",  
+      ylab="Fitted Probability of Illness")
```





Q: Why does the plot show a negative slope? Does it make sense?

## Grouped Data: Binomial Response

If several observations have the same  $\mathbf{x}$  (“replications”), then they have the same  $\pi(\mathbf{x})$ .

Summing binary (0/1) observations with the same  $\mathbf{x}$  gives **grouped** data:

$$Y_i \sim \text{binomial}(n_i, \pi(\mathbf{x}_i))$$

where “ $i$ ” now refers to the  $i$ th group (of  $n_i$  binary obs.).

Note: Both  $Y_i$  and  $n_i$  (or  $n_i - Y_i$ ) must be included in the data.

## Remarks:

- ▶ Whether data are grouped or ungrouped, fitting with maximum likelihood gives the same results.

## R Example: Snoring & Heart Disease Data

```
> snoreheart <- read.table("snoreheart.txt", header=TRUE)
```

```
> snoreheart
```

	Disease	NoDisease	Snoring
1	24	1355	0
2	35	603	2
3	21	192	4
4	30	224	5

Remark: Snoring level is actually ordinal, but we convert it to numerical scores, as suggested by Agresti.

```

> snorefit <- glm(cbind(Disease, NoDisease) ~ Snoring, family=binomial,
+               data=snoreheart)

> summary(snorefit)
...

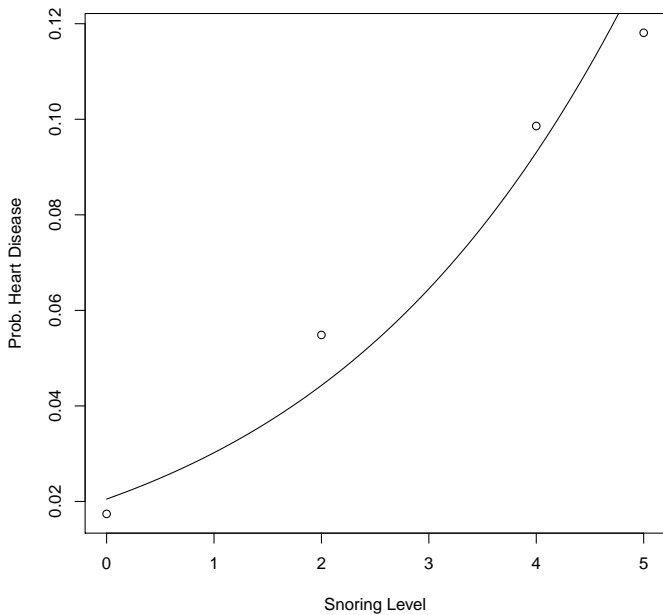
Coefficients:
              Estimate Std. Error z value Pr(>|z|)
(Intercept) -3.86625    0.16621  -23.261  < 2e-16 ***
Snoring      0.39734    0.05001   7.945 1.94e-15 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

...

```

Note: The response must include both the “successes” (Disease) and “failures” (NoDisease).

```
> with(snoreheart, plot(Snoring, Disease/(Disease+NoDisease),  
+                       xlab="Snoring Level", ylab="Prob. Heart Disease"))  
  
> curve(predict(snorefit, data.frame(Snoring=x), type="response"), add=TRUE)
```



## For $2 \times 2$ Tables

$x = 1$	$Y_1$	$n_1 - Y_1$	$n_1$
$x = 0$	$Y_2$	$n_2 - Y_2$	$n_2$

Note: Can regard as grouped data with two groups.

A binomial regression model (with  $x = 0$  or  $1$ ) is equivalent to the independent binomial model:

$$\left. \begin{array}{l} Y_1 \sim \text{binomial}(n_1, \pi_1 = \pi(1)) \\ Y_2 \sim \text{binomial}(n_2, \pi_2 = \pi(0)) \end{array} \right\} \text{ independent}$$



For **logistic** regression:

$$\text{logit}(\pi(x)) = \alpha + \beta x$$

so the odds ratio is

$$\begin{aligned}\theta &= \frac{\pi_1/(1 - \pi_1)}{\pi_2/(1 - \pi_2)} = \exp(\text{logit}(\pi_1) - \text{logit}(\pi_2)) \\ &= \exp(\alpha + \beta \cdot 1 - (\alpha + \beta \cdot 0)) = e^\beta\end{aligned}$$

So  $\beta$  is the log odds ratio.

## Other Links

Let  $F$  be a continuous and invertible c.d.f. on the real line.

A reasonable link might be

$$g(\pi) = F^{-1}(\pi)$$

since it transforms interval  $(0, 1)$  to the whole real line.

Using the c.d.f.  $\Phi$  for a standard normal is called **probit regression**.

It relates to the concept of latent variables — see Agresti, Sec. 4.2.6.

# Count Response

For binomial data, the maximum possible count is known (for each observation).

What if there are no known maximum counts?

Counts of independently-occurring incidents (without any maximum) are often modeled using the Poisson distribution ...

$$Y_i \sim \text{Poisson}(\mu(\mathbf{x}_i))$$

$$\text{E}(Y_i) = \mu(\mathbf{x}_i) \quad \text{var}(Y_i) = \mu(\mathbf{x}_i)$$

Recall canonical link:

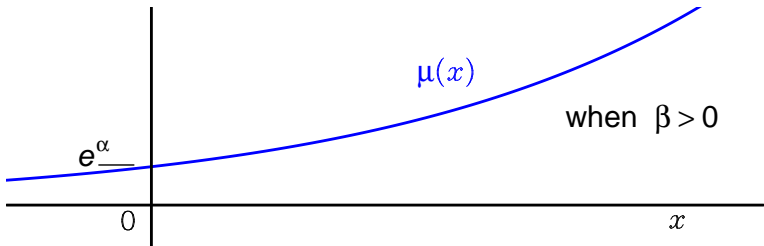
$$\ln \mu(\mathbf{x}_i) = \alpha + \beta_1 x_{i1} + \cdots + \beta_p x_{ip}$$

**(loglinear model)**

When  $p = 1$ ,

$$\ln \mu(x) = \alpha + \beta x$$

$$\Leftrightarrow \mu(x) = e^{\alpha + \beta x} = e^{\alpha} (e^{\beta})^x$$



## R Example: Horseshoe Crab Data

$Y_i$  = number of males (“satellites”) by female  $i$

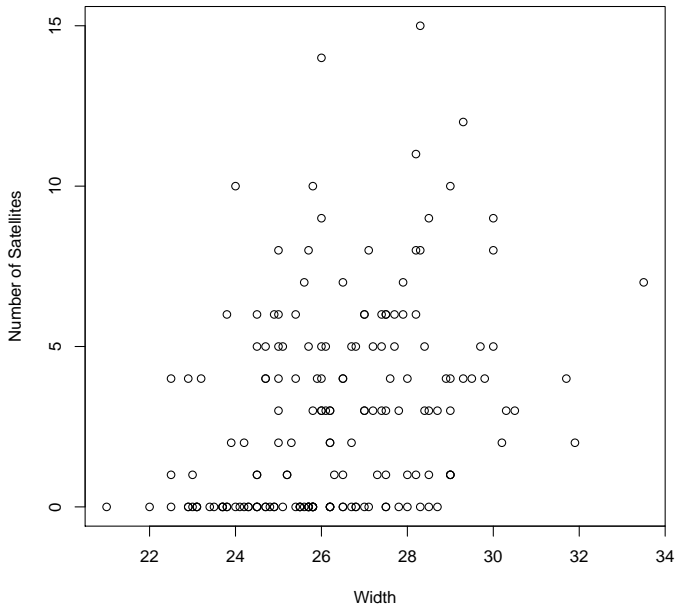
$x_i$  = width (cm) of female  $i$

```
> horseshoe <- read.table("horseshoe.txt", header=TRUE)
```

```
> head(horseshoe)
```

	color	spine	width	satell	weight	y
1	3	3	28.3	8	3050	1
2	4	3	22.5	0	1550	0
3	2	1	26.0	9	2300	1
4	4	3	24.8	0	2100	0
5	4	3	26.0	4	2600	1
6	3	3	23.8	0	2100	0

```
> plot(satell ~ width, data=horseshoe, xlab="Width",  
+       ylab="Number of Satellites")
```



```
> hsfit <- glm(satell ~ width, family=poisson, data=horseshoe)
```

```
> summary(hsfit)
```

```
...
```

Coefficients:

	Estimate	Std. Error	z value	Pr(> z )	
(Intercept)	-3.30476	0.54224	-6.095	1.1e-09	***
width	0.16405	0.01997	8.216	< 2e-16	***

```
---
```

Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1

(Dispersion parameter for poisson family taken to be 1)

Null deviance: 632.79 on 172 degrees of freedom  
Residual deviance: 567.88 on 171 degrees of freedom  
AIC: 927.18

Number of Fisher Scoring iterations: 6



Log-link is used by default:

$$\hat{\mu}(x) \approx e^{-3.305+0.164x}$$

e.g. a 30 cm female should have about  $e^{-3.305+0.164 \times 30} \approx 5.03$  satellites

For each 1 cm increase in width, the mean number of satellites increases by a factor of

$$e^{0.164} \approx 1.18 \quad (18\%)$$

## Rate Models

$E(Y_i) = \mu_i$  is sometimes expected to be proportional to another observed variable  $t_i > 0$ :

$$\mu_i = \lambda_i t_i$$

e.g.

$Y_i$  = cases of rare disease in nation  $i$

$t_i$  = national population (known)

$\lambda_i$  = disease **rate** (unknown)

( $t$  could alternatively be a temporal or spatial extent)

If

$$\ln \lambda_i = \alpha + \beta_1 x_{i1} + \cdots + \beta_p x_{ip}$$

then

$$\begin{aligned}\ln \mu_i &= \ln \lambda_i + \ln t_i \\ &= \alpha + \beta_1 x_{i1} + \cdots + \beta_p x_{ip} + \ln t_i\end{aligned}$$

Note:  $\ln t_i$  has no coefficient.

Call  $\ln t_i$  an **offset**.

## R Example: British Train Collisions

$Y_i$  = number of collisions b/w trains & road vehicles (year  $i$ )

$x_i$  = year minus 1975

$t_i$  = total km of train travel (millions)

```
> tc <- read.table("traincollisions.txt", header=TRUE)
```

```
> head(tc)
```

	Year	KM Train	TrRd	
1	2003	518	0	3
2	2002	516	1	3
3	2001	508	0	4
4	2000	503	1	3
5	1999	505	1	2
6	1998	487	0	4

```
> tcfit <- glm(TrRd ~ I(Year-1975), offset = log(KM), family=poisson, data=tc)
> summary(tcfite)
```

...

Coefficients:

	Estimate	Std. Error	z value	Pr(> z )
(Intercept)	-4.21142	0.15892	-26.50	< 2e-16 ***
I(Year - 1975)	-0.03292	0.01076	-3.06	0.00222 **

---

Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1

(Dispersion parameter for poisson family taken to be 1)

Null deviance: 47.376 on 28 degrees of freedom  
 Residual deviance: 37.853 on 27 degrees of freedom  
 AIC: 133.52

Number of Fisher Scoring iterations: 5

```
> plot(1000*TrRd/KM ~ Year, data=tc,  
+       ylab="Collisions per Billion Train-Kilometers")  
  
> curve(1000*predict(tcfit, data.frame(Year=x,KM=1), type="response"),  
+       add=TRUE)
```

(Note: Set rate variable  $t$  to 1 to get estimates of the rate itself.)

