Week 2 Notes: Basics of Generalized Linear Models

Antonio R. Linero
University of Texas at Austin

Historical Motivation: Linear Model

Ordinary Linear Regression:

$$Y_i = X_i^\top \beta + \epsilon_i, \qquad \epsilon_i \sim \text{Normal}(0, \sigma^2).$$

Strengths and limitations?

Historical Motivation: Linear Model

Ordinary Linear Regression:

$$Y_i = X_i^\top \beta + \epsilon_i, \qquad \epsilon_i \sim \text{Normal}(0, \sigma^2).$$

Strengths and limitations?

- Computationally simple
- Mostly effective
- Cannot handle non-numeric data, which is usually heteroskedastic! (Why important?)

Exercise: Variance Stabilizing Transformations

Suppose that $Y \sim \text{Poisson}(\lambda)$. Using the expansion

$$g(y) \approx g(\lambda) + (y - \lambda)\, g'(\lambda)$$

find a transformation $g(\cdot)$ such that the variance of $g(Y)$ is approximately constant.

Transformation Models

Exercise: Variance Stabilizing Transformations

Suppose that $Y \sim \text{Poisson}(\lambda)$. Using the expansion

$$g(y) \approx g(\lambda) + (y - \lambda)\, g'(\lambda)$$

find a transformation $g(\cdot)$ such that the variance of $g(Y)$ is approximately constant.

Old people model: $\sqrt{Y_i} = X_i^\top \beta + \epsilon_i$ when Y_i is count valued.
Limitations?

Overall Goal:

GLMs are a **computationally tractable generalization of the linear model to new outcomes!**

1. The *stochastic component:* a choice of model for Y_i (e.g., binomial, normal, gamma, Poisson)

Overall Goal:

GLMs are a **computationally tractable generalization of the linear model to new outcomes!**

1. The *stochastic component:* a choice of model for Y_i (e.g., binomial, normal, gamma, Poisson)
2. The *systematic component:* $\eta_i = X_i^\top \beta$, quantifying the effect of predictors.

Overall Goal:

GLMs are a **computationally tractable generalization of the linear model to new outcomes!**

1. The *stochastic component:* a choice of model for Y_i (e.g., binomial, normal, gamma, Poisson)
2. The *systematic component:* $\eta_i = X_i^\top \beta$, quantifying the effect of predictors.
3. The *link function:* tells the model how the previous two components talk to each other by setting $g\left(\mathbb{E}(Y_i \mid X_i)\right) = \eta_i$.

Exponential Dispersion Families

Definition (Exponential Dispersion Family)

A family of distributions $\{f(\cdot;\theta,\phi) : \theta \in \Theta, \phi \in \Phi\}$ is an *exponential dispersion family* if we can write

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

for some *known* functions $b(\cdot)$ and $c(\cdot,\cdot)$. The parameter θ is referred to as the *canonical parameter* of the family and ϕ is referred to as the *dispersion parameter*.

Examples

Exercise: Examples of Exponential Dispersion Families

Show that the following families are types of exponential dispersion families, and find the corresponding b, c, θ, ϕ.

1. $Y \sim \text{Normal}(\mu, \sigma^2)$
2. $Y = Z/N$ where $Z \sim \text{Binomial}(N, p)$
3. $Y \sim \text{Poisson}(\lambda)$
4. $Y \sim \text{Gam}(\alpha, \beta)$ (parameterized so that $\mathbb{E}(Y) = \alpha/\beta$).

GLMs

Definition: Generalized Linear Models

Suppose that we have $\mathcal{D} = \{(Y_i, x_i) : i = 1, \dots, N\}$ (with the x_i's regarded as fixed constants). We say that the Y_i's follow a *generalized linear model* if:

1. Y_i has density/mass function

$$f(y_i \mid \theta_i, \phi/\omega_i) = \exp\left\{\frac{y_i\,\theta_i - b(\theta_i)}{\phi/\omega_i} + c(y_i; \phi/\omega_i)\right\}$$

   where the coefficients $\omega_1, \dots, \omega_N$ are known. This is referred to as the *stochastic component* of the model.

2. For some known (invertible) *link function* $g(\mu)$ we have

$$g(\mu_i) = x_i^\top \beta$$

   where $\mu_i = \mathbb{E}(Y_i \mid \theta_i, \phi/\omega_i)$. This is referred to as the *systematic component* of the model. The term $\eta_i = x_i^\top \beta$ is known as the *linear predictor*.

Moments

Exercise: GLM Moments

Suppose that $Y \sim f(y; \theta, \phi/\omega)$ for some exponential dispersion family. Show that

1. $\mathbb{E}(Y \mid \theta, \phi/\omega) = b'(\theta)$; and
2. $\mathrm{Var}(Y \mid \theta, \phi/\omega) = \frac{\phi}{\omega} b''(\theta)$.

Moments

Exercise: GLM Moments

Suppose that $Y \sim f(y;\theta,\phi/\omega)$ for some exponential dispersion family. Show that

1. $\mathbb{E}(Y \mid \theta,\phi/\omega) = b'(\theta)$; and
2. $\mathrm{Var}(Y \mid \theta,\phi/\omega) = \frac{\phi}{\omega} b''(\theta)$.

The function $V(\mu) = b''\{(b')^{-1}(\mu)\}$ is called the variance function.

Moments

Exercise: GLM Moments

Suppose that $Y \sim f(y; \theta, \phi/\omega)$ for some exponential dispersion family. Show that

1. $\mathbb{E}(Y \mid \theta, \phi/\omega) = b'(\theta)$; and
2. $\mathrm{Var}(Y \mid \theta, \phi/\omega) = \frac{\phi}{\omega} b''(\theta)$.

The function $V(\mu) = b''\{(b')^{-1}(\mu)\}$ is called the variance function.

- **Binomial**: $V(\mu) = \mu(1 - \mu)$
- **Poisson**: $V(\mu) = \mu$
- **Gamma**: $V(\mu) = \mu^2$

Canonical Links

Definition: Canonical Link Function

The *canonical link* takes $g(\mu) = (b')^{-1}(\mu)$. By definition this gives the model

$$f(y_i \mid x_i, \omega_i, \theta, \phi) = \exp\left\{ \frac{y_i x_i^\top \beta - b(x_i^\top \beta)}{\phi/\omega_i} + c(y_i; \phi/\omega_i) \right\},$$

i.e., we use the exponential dispersion family with $\theta_i = x_i^\top \beta$.

Canonical Links

Definition: Canonical Link Function

The *canonical link* takes $g(\mu) = (b')^{-1}(\mu)$. By definition this gives the model

$$f(y_i \mid x_i, \omega_i, \theta, \phi) = \exp\left\{\frac{y_i x_i^\top \beta - b(x_i^\top \beta)}{\phi/\omega_i} + c(y_i; \phi/\omega_i)\right\},$$

i.e., we use the exponential dispersion family with $\theta_i = x_i^\top \beta$.

a. $Y \sim \text{Normal}(\mu, \sigma^2)$: $g(\mu) = \mu$.

Canonical Links

Definition: Canonical Link Function

The *canonical link* takes $g(\mu) = (b')^{-1}(\mu)$. By definition this gives the model

$$f(y_i \mid x_i, \omega_i, \theta, \phi) = \exp\left\{ \frac{y_i x_i^\top \beta - b(x_i^\top \beta)}{\phi/\omega_i} + c(y_i; \phi/\omega_i) \right\},$$

i.e., we use the exponential dispersion family with $\theta_i = x_i^\top \beta$.

a. $Y \sim \text{Normal}(\mu, \sigma^2)$: $g(\mu) = \mu$.
b. $Y \sim \text{Poisson}(\lambda)$: $g(\mu) = \log \mu$.

Canonical Links

Definition: Canonical Link Function

The *canonical link* takes $g(\mu) = (b')^{-1}(\mu)$. By definition this gives the model

$$f(y_i \mid x_i, \omega_i, \theta, \phi) = \exp\left\{\frac{y_i x_i^\top \beta - b(x_i^\top \beta)}{\phi/\omega_i} + c(y_i; \phi/\omega_i)\right\},$$

i.e., we use the exponential dispersion family with $\theta_i = x_i^\top \beta$.

a. $Y \sim \text{Normal}(\mu, \sigma^2)$: $g(\mu) = \mu$.
b. $Y \sim \text{Poisson}(\lambda)$: $g(\mu) = \log \mu$.
c. $Y = Z/n$ with $Z \sim \text{Binomial}(n, p)$: $g(\mu) = \log\{\mu/(1-\mu)\}$.

Canonical Links

Definition: Canonical Link Function

The *canonical link* takes $g(\mu) = (b')^{-1}(\mu)$. By definition this gives the model

$$f(y_i \mid x_i, \omega_i, \theta, \phi) = \exp\left\{ \frac{y_i x_i^\top \beta - b(x_i^\top \beta)}{\phi/\omega_i} + c(y_i; \phi/\omega_i) \right\},$$

i.e., we use the exponential dispersion family with $\theta_i = x_i^\top \beta$.

a. $Y \sim \text{Normal}(\mu, \sigma^2)$: $g(\mu) = \mu$.
b. $Y \sim \text{Poisson}(\lambda)$: $g(\mu) = \log \mu$.
c. $Y = Z/n$ with $Z \sim \text{Binomial}(n, p)$: $g(\mu) = \log\{\mu/(1-\mu)\}$.
d. $Y \sim \text{Gam}(\alpha, \beta)$: $g(\mu) = -1/\mu$.

Fitting GLMs in R

```
my_glm <- glm(
  response ~ predictor_1 + predictor_2 + and_so_forth,
  data = my_data,
  family = my_family
)
```

- `family`: what type of GLM? (`poisson`, `binomial`, `gamma`, `binomial("probit")`)
- `data`: what dataset?
- Uses R formula specification syntax (see reference material or Google)

Fitting Bayesian GLMs in R

```r
library(rstanarm)
my_glm <- stan_glm(
  response ~ predictor_1 + predictor_2 +  and_so_forth,
  data = my_data,
  family = my_family
)
```

- Same syntax, more-or-less
- Uses "default" priors! I guess you might want to change these…

Logistic Regression

Logistic regression:

$$Y_i = Z_i/n_i \qquad \text{where} \qquad Z_i \sim \text{Binomial}(n_i, p_i).$$

with the canonical link

$$p_i = \frac{\exp(x_i^\top \beta)}{1 + \exp(x_i^\top \beta)} \iff \text{logit}(p_i) = X_i^\top \beta$$

What the Coefficients Represent

If

$$\frac{\text{Odds}(Y_i = 1 \mid X_i)}{\text{Odds}(Y_{i'} = 1 \mid X_{i'})} = \frac{\Pr(Y_i = 1 \mid X_i)\ \Pr(Y_{i'} = 0 \mid X_{i'})}{\Pr(Y_i = 0 \mid X_i)\ \Pr(Y_{i'} = 1 \mid X_{i'})}$$

then the odds ratio is given by $e^{\beta_2 \delta}$ if X_i and $X_{i'}$ are identical except that $X_{i2} = X_{i'2} + \delta$.

Challenger

FlightNumber	Temperature	Pressure	Fail	nFailures	Damage
1	66	50	0	0	0
2	70	50	1	1	4
3	69	50	0	0	0
5	68	50	0	0	0
6	67	50	0	0	0
7	72	50	0	0	0

Goal: should stakeholders have been able to predict the failure of the O-rings on the challenger? *If we repeated the Challenger launch under similar conditions, what would the probability of O-ring failure be?*

Model

Simple logistic regression model:

$$\text{logit}(p_i) = \beta_0 + \beta_{\text{temp}} \times \text{temp}_i.$$

In `R` we can fit the this model by maximum likelihood as follows.

```
challenger_fit <- glm(
  Fail ~ Temperature,
  data = challenger,
  family = binomial
)
```

Summary

```
summary(challenger_fit)
```

```
##
## Call:
## glm(formula = Fail ~ Temperature, family = binomial, data = challenger)
##
## Coefficients:
##             Estimate Std. Error z value Pr(>|z|)
## (Intercept)  15.0429     7.3786   2.039   0.0415 *
## Temperature  -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
```

How do we interpret the output here?

Other Functions

```
coef(challenger_fit)
```

```
## (Intercept) Temperature
##  15.0429016  -0.2321627
```

Other Functions

```
coef(challenger_fit)
```

```
## (Intercept) Temperature
##  15.0429016  -0.2321627
```

```
confint(challenger_fit)
```

```
## Waiting for profiling to be done...

##                  2.5 %      97.5 %
## (Intercept)  3.3305848 34.34215133
## Temperature -0.5154718 -0.06082076
```

Other Functions

```
coef(challenger_fit)
```

```
## (Intercept) Temperature
##  15.0429016  -0.2321627
```

```
confint(challenger_fit)
```

```
## Waiting for profiling to be done...
```

```
##                  2.5 %      97.5 %
## (Intercept)  3.3305848 34.34215133
## Temperature -0.5154718 -0.06082076
```

```
vcov(challenger_fit)
```

```
##             (Intercept) Temperature
## (Intercept)  54.4441826 -0.79638547
## Temperature  -0.7963855  0.01171512
```

Challenger Predictions

What is the MLE of the probability of failure at different temperatures?

```
predict(challenger_fit,
        newdata = data.frame(Temperature = c(40, 50, 60)),
        type = 'response',
        se.fit = TRUE)
```

```
## $fit
##         1         2         3
## 0.9968475 0.9687735 0.7527135
##
## $se.fit
##           1           2           3
## 0.009674669 0.061205420 0.190948130
##
## $residual.scale
## [1] 1
```

Fitting the Bayesian Version

A Bayesian version can also be fit as follows.

```
challenger_bayes <- rstanarm::stan_glm(
  Fail ~ Temperature,
  data = challenger,
  family = binomial
)
```

Fitting the Bayesian Version

A Bayesian version can also be fit as follows.

```
challenger_bayes <- rstanarm::stan_glm(
  Fail ~ Temperature,
  data = challenger,
  family = binomial
)
```

Using the Bayesian version, let's plot the samples of the function

$$f(\texttt{temp}) = \{1 + \exp(-\beta_0 - \beta_1 \texttt{temp})\}^{-1}.$$

Code

```r
## For Reproducibility
set.seed(271985)

## Converts the rstanarm object to a matrix
beta_samples <- as.matrix(challenger_bayes)

## Some Colors
pal <- ggthemes::colorblind_pal()(8)

## Set up plotting region
plot(
  x = challenger$Temperature,
  y = challenger$Fail,
  ylab = "Failure?",
  xlab = "Temperature",
  type = 'n'
)

## A function for adding estimate
plot_line <- function(beta, col = 'gray') {
  plot(function(x) 1 / (1 + exp(-beta[1] - beta[2] * x)),
       col = col, add = TRUE, xlim = c(40, 90), n = 200)
}

## Apply plot_line for a random collection of betas
tmpf <- function(i) plot_line(beta_samples[i,])
tmp <- sample(1:4000, 200) %>% lapply(tmpf)

## Get the Bayes estimate of the probability
tempgrid <- seq(from = 40, to = 90, length = 200)
bayes_est <- predict(challenger_bayes,
  type = 'response',
  newdata = data.frame(Temperature = tempgrid)
)
lines(tempgrid, bayes_est, col = pal[3], lwd = 4)
```

Results

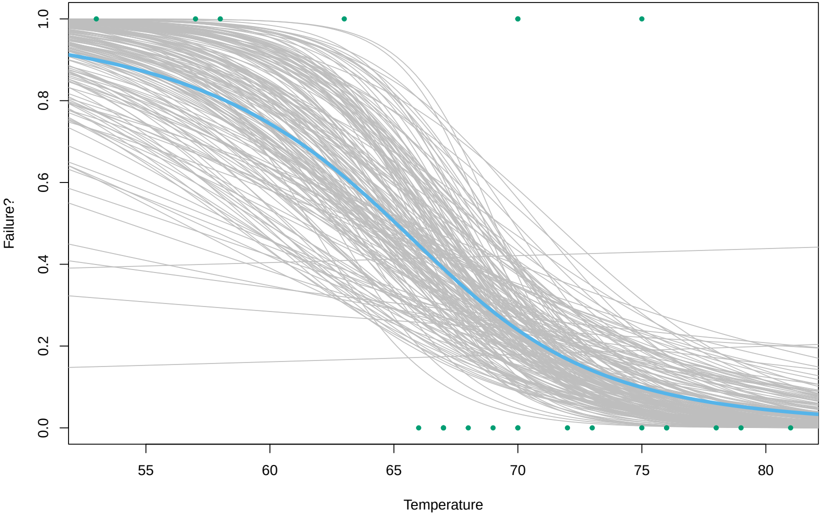

Figure 1: Posterior samples of the probability of failure.

Predictions

```r
predict(challenger_bayes,
        newdata = data.frame(Temperature = 30),
        type = 'response')
```

```
##         1
## 0.9857381
```

Bayesian believes that the shuttle will experience an O-ring failure with probability roughly 98%.

Poisson Log-Linear Models

For count data:

$$Y_i \sim \text{Poisson}(\mu_i) \qquad \text{where} \qquad \log(\mu_i) = x_i^\top \beta.$$

This is referred to as a *Poisson log-linear model*. Equivalently, we have $\mu_i = \exp(x_i^\top \beta)$.

Poisson Log-Linear Models

For count data:

$$Y_i \sim \text{Poisson}(\mu_i) \qquad \text{where} \qquad \log(\mu_i) = x_i^\top \beta.$$

This is referred to as a *Poisson log-linear model.* Equivalently, we have $\mu_i = \exp(x_i^\top \beta)$.

Poisson distribution models the *number of times an event occurs in a given time, or within a given space.*

- Number of homicides in a city
- Number of goals scored in a soccer game
- Take values 0, 1,2 , . . . with no obvious upper bound.

For example, it might be used to model the

Coefficients

Exercise: Coefficients in a Poisson Regression

Suppose we fit a Poisson log-linear model $\log(\mu_i) = \beta_0 + \beta_{i1}X_{i1} + \beta_{i2}X_{i2}$. Show that a change in X_{i2} by δ units, holding X_{i1} fixed, results in a *multiplicative effect on the mean*:

$$\mu_{\text{new}} = e^{\beta_2 \delta} \mu_{\text{old}}$$

Ships Dataset (McCullaugh and Nelder)

```r
ships <- MASS::ships
head(ships)
```

```
##   type year period service incidents
## 1    A   60     60     127         0
## 2    A   60     75      63         0
## 3    A   65     60    1095         3
## 4    A   65     75    1095         4
## 5    A   70     60    1512         6
## 6    A   70     75    3353        18
```

- `type`: type of vessel
- `year`: year the vessel was constructed
- `period`: time period vessel is operating other
- `service`: number of months of service of ships of this type
- `incidents`: total number of incidents

Questions

1. Do certain types of ships tends to have higher numbers of incidents, after controlling for other factors?
2. Were some periods more prone to other incidents, after controlling for other factors?
3. Did ships built in certain years have more accidents than others?

A Simple Loglinear Model

Set $\texttt{incidents}_i \sim \text{Poisson}(\mu_i)$ with

$$\log \mu_i = \beta_0 + \beta_{\texttt{service}} \cdot \texttt{service}_i + \beta_{\texttt{type}} \cdot \texttt{type}_i + \beta_{\texttt{period}} \cdot \texttt{period}_i + \beta_{\texttt{year}} \cdot \texttt{year}_i.$$

A Simple Loglinear Model

Set $\texttt{incidents}_i \sim \text{Poisson}(\mu_i)$ with

$$\log \mu_i = \beta_0 + \beta_{\texttt{service}} \cdot \texttt{service}_i + \beta_{\texttt{type}} \cdot \texttt{type}_i + \beta_{\texttt{period}} \cdot \texttt{period}_i + \beta_{\texttt{year}} \cdot \texttt{year}_i.$$

Slightly better model:

$$\log \mu_i = \beta_0 + \log(\texttt{service}_i) + \beta_{\texttt{type}} \cdot \texttt{type}_i + \beta_{\texttt{period}} \cdot \texttt{period}_i + \beta_{\texttt{year}} \cdot \texttt{year}_i.$$

Term $\log(\texttt{service}_i)$ is called an *offset*. Why is this better?

Fitting Ships

```
ships_glm <- glm(
  incidents ~ type + factor(period) + factor(year),
  family = poisson,
  offset = log(service),
  data = dplyr::filter(ships, service > 0)
)

print(summary(ships_glm))
```

Fitting Ships

```
##
## Call:
## glm(formula = incidents ~ type + factor(period) + factor(year),
##     family = poisson, data = dplyr::filter(ships, service > 0),
##     offset = log(service))
##
## Coefficients:
##                   Estimate Std. Error z value Pr(>|z|)
## (Intercept)       -6.40590    0.21744 -29.460  < 2e-16 ***
## typeB             -0.54334    0.17759  -3.060  0.00222 **
## typeC             -0.68740    0.32904  -2.089  0.03670 *
## typeD             -0.07596    0.29058  -0.261  0.79377
## typeE              0.32558    0.23588   1.380  0.16750
## factor(period)75   0.38447    0.11827   3.251  0.00115 **
## factor(year)65     0.69714    0.14964   4.659 3.18e-06 ***
## factor(year)70     0.81843    0.16977   4.821 1.43e-06 ***
## factor(year)75     0.45343    0.23317   1.945  0.05182 .
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for poisson family taken to be 1)
##
##     Null deviance: 146.328  on 33  degrees of freedom
## Residual deviance:  38.695  on 25  degrees of freedom
## AIC: 154.56
##
## Number of Fisher Scoring iterations: 5
```

Discuss identifiability of coefficients.

Conclusions

- Strong evidence for effects of `period`: `period75` has more incidents per month of service.
- Incidents in year 60 seem relatively low (quite different from 65 and 70, some evidence of fewer incidents in 75 as well), all other things being equal.
- Evidence for differences across types of ships, with (for example) B having fewer incidents than A.

Exercises

Exercise: Bayesian Poisson Loglinear Model

Fit this function using `stan_glm`, then try out the `plot` function for `stanreg` objects. Describe your results.

Exercises

Exercise: Overdispersion

A problem with Poisson log-linear models is that they impose the restriction $\mathbb{E}(Y_i) = \mathrm{Var}(Y_i)$ so that the variance is completely constrained by the mean. Count data is referred to as *overdispersed* if $\mathrm{Var}(Y_i) > \mathbb{E}(Y_i)$.

a. Consider the model $Y \sim \mathrm{Poisson}(\lambda)$ (given λ) and $\lambda \sim \mathrm{Gam}(k, k/\mu)$. Find the mean and variance of Y. Is Y overdispersed?

b. Show that Y marginally has a negative binomial distribution with k failures and success probability $\mu/(k+\mu)$; recall that the negative binomial distribution has mass function

$$f(y \mid k, p) = \binom{k+y-1}{y} p^y (1-p)^k.$$

c. The following data is taken from Table 14.6 in Categorical Data Analysis, 3rd edition, by Alan Agresti.

```r
poisson_data <- data.frame(
  Response = 0:6,
  Black = c(119,16,12,7,3,2,0),
  White = c(1070,60,14,4,0,0,1)
)
knitr::kable(poisson_data, booktabs = TRUE)
```

Response	Black	White
0	119	1070
1	16	60
2	12	14
3	7	4
4	3	0
5	2	0
6	0	1