

# Generalized Linear Model & Logistic Regression

## Fundamental Techniques in Data Science



**Utrecht  
University**

Kyle M. Lang

Department of Methodology & Statistics  
Utrecht University

# Outline

---

## Generalized Linear Model

Complex theme, will not consolidate at first.

## Logistic Regression

## Classification



# General Linear Model

---

So far, we've been discussing models with this form:

$$Y = \beta_0 + \sum_{p=1}^P \beta_p X_p + \varepsilon$$

This type of model is known as the *general linear model*.

- All flavors of linear regression are general linear models.
  - SLR, MLR
  - t-test, ANOVA, ANCOVA
  - Multilevel linear regression models



# Components of the General Linear Model

We can break our model into pieces:

$$\eta = \beta_0 + \sum_{p=1}^P \beta_p X_p$$

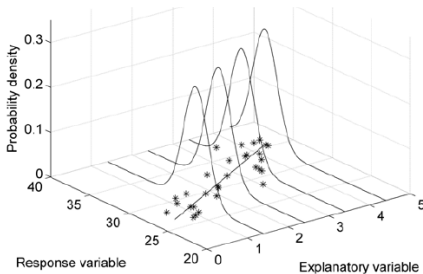
$$Y = \eta + \varepsilon$$

$\varepsilon \sim N(0, \sigma^2)$ , so we can also write:

$$Y \sim N(\eta, \sigma^2)$$

Where:

- $\eta$  is the *systematic component* of the model (AKA, the *linear predictor*).
- The normal distribution,  $N(\cdot, \cdot)$ , is the model's *random component*.



# Components of the General Linear Model

---

The purpose of general linear modeling (i.e., regression modeling) is to build a model of the outcome's mean,  $\mu_Y$ .

- In this case,  $\mu_Y = \eta$ .
- The systematic component defines the mean of  $Y$ .

The random component quantifies variability around  $\mu_Y$  (i.e., error variance).

- In the general linear model, we assume that this error variance follows a normal distribution.



# GENERALIZED LINEAR MODEL



# Extending the General Linear Model

---

We can generalize the models we've been using in two important ways:

1. Allow for random components other than the normal distribution.
2. Allow for more complicated relations between  $\mu_Y$  and  $\eta$ .
  - Allow:  $g(\mu_Y) = \eta$

These extensions lead to the class of *generalized linear models* (GLMs).

generalized linear models  $\neq$  general linear model (GLM)



# Components of the Generalized Linear Model

The random component in a GLM can be any distribution from the so-called *exponential family*.

- The exponential family contains many popular distributions:
  - Normal
  - Binomial
  - Poisson
  - Many others...

The systematic component of a GLM is exactly the same as it is in general linear models:

$$\eta = \beta_0 + \sum_{p=1}^P \beta_p X_p$$





# Link Functions

---

In GLMs,  $\eta$  does not directly describe  $\mu_Y$ .

- We first transform  $\mu_Y$  via a *link function*.
- $g(\mu_Y) = \eta$

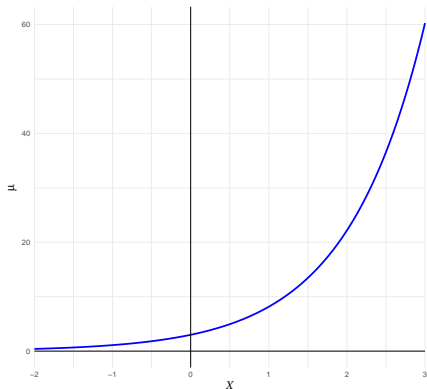
The link function performs two important functions.

1. Linearize the association between  $\mathbf{X}$  and  $Y$ .
  - Nonlinear:  $\mathbf{X} \rightarrow \mu_Y$
  - Linear:  $\mathbf{X} \rightarrow g(\mu_Y)$
2. Allows GLMs for outcomes with restricted ranges without requiring any restrictions on the range of the  $\{X_p\}$ .
  - In many cases,  $\mu_Y$  has a limited range.
    - Counts:  $\mu_Y > 0$
    - Probabilities:  $\mu_Y \in [0, 1]$
  - When correctly specified,  $g(\mu_Y)$  can take any value on the real line.



# Visualizing Link Functions

Raw Conditional Mean



Linearized Conditional Mean

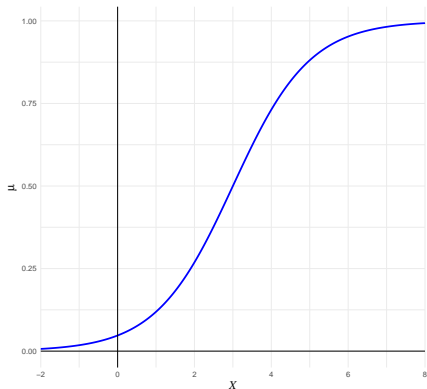


Allowed for negative values

# Visualizing Link Functions

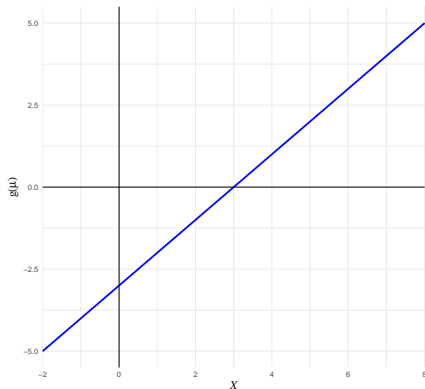
"Sigmoidal function, such a nice word to say, I encourage you to use it as frequent as possible."

Raw Conditional Mean



Logit is the link function for logistic reg

Linearized Conditional Mean



# Components of the Generalized Linear Model

Every GLM is built from three components:

1. The systematic component,  $\eta$ .
  - A linear function of the predictors,  $\{X_p\}$ .
  - Describes the association between  $\mathbf{X}$  and  $Y$ .
2. The link function,  $g(\mu_Y)$ .
  - Linearizes the relation between  $\mathbf{X}$  and  $Y$ .
  - Transforms  $\mu_Y$  so that it can take any value on the real line.
3. The random component,  $P(Y|g^{-1}(\eta))$ 
  - The distribution of the observed  $Y$ .
  - Quantifies the error variance around  $\eta$ .



# General Linear Model as a Special Case

---

The general linear model is a special case of GLM.

1. Systematic component:

$$\eta = \beta_0 + \sum_{p=1}^P \beta_p X_p$$

2. Link function:

$$\mu_Y = \eta$$

3. Random component:

$$Y \sim N(\eta, \sigma^2)$$



# Example

---

```
data(iris)

## General linear model:
lmFit <- lm(Petal.Length ~ Petal.Width + Species, data = iris)

## Generalized linear model:
glmFit <- glm(Petal.Length ~ Petal.Width + Species,
              family = gaussian(link = "identity"),
              data = iris)
```

family of distributions

# Example

---

```
partSummary(lmFit, 2)
```

```
## Residuals:
```

##	Min	1Q	Median	3Q	Max
##	-1.02977	-0.22241	-0.01514	0.18180	1.17449

```
partSummary(glmFit, 2)
```

```
## Coefficients:
```

##	Estimate	Std. Error	t value	Pr(> t )
## (Intercept)	1.21140	0.06524	18.568	< 2e-16
## Petal.Width	1.01871	0.15224	6.691	4.41e-10
## Speciesversicolor	1.69779	0.18095	9.383	< 2e-16
## Speciesvirginica	2.27669	0.28132	8.093	2.08e-13

# Example

---

```
partSummary(lmFit, 3)
```

```
## Coefficients:
```

##	Estimate	Std. Error	t value	Pr(> t )
## (Intercept)	1.21140	0.06524	18.568	< 2e-16
## Petal.Width	1.01871	0.15224	6.691	4.41e-10
## Speciesversicolor	1.69779	0.18095	9.383	< 2e-16
## Speciesvirginica	2.27669	0.28132	8.093	2.08e-13

```
partSummary(glmFit, 3)
```

```
## (Dispersion parameter for gaussian family taken to be 0.1426948)
```

General linear is a special case of  
generalized linear models.



# LOGISTIC REGRESSION



# Logistic Regression

---

So why do we care about the GLM when linear regression models have worked thus far?

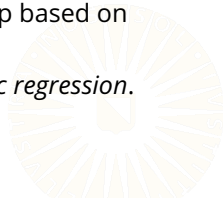
- In a word: Classification.

In the classification task, we have a discrete, qualitative outcome.

- We will begin with the situation of two-level outcomes.
  - Alive or Dead
  - Pass or Fail
  - Pay or Default

We want to build a model that predicts class membership based on some set of interesting features.

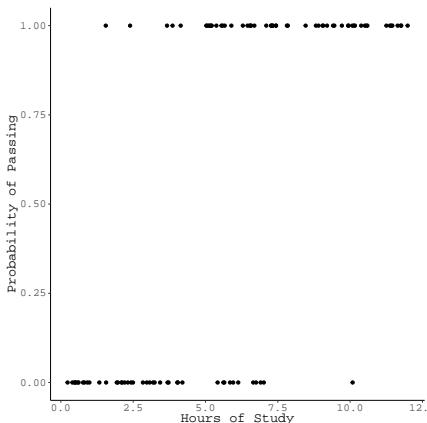
- To do so, we will use a very useful type of GLM: *logistic regression*.



# Classification Example

Suppose we want to know the effect of study time on the probability of passing an exam.

- The probability of passing must be between 0 and 1.
- We care about the probability of passing, but we only observe absolute success or failure.
  - $Y \in \{1, 0\}$



Text

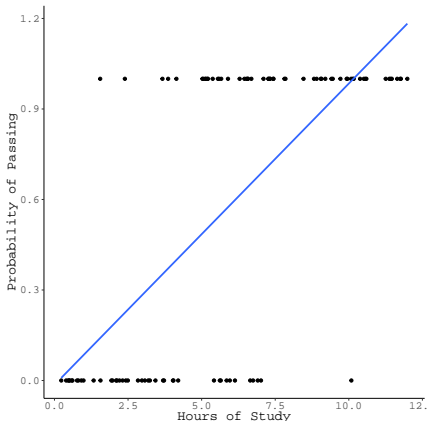
# Linear Regression for Binary Outcomes?

Problems:

- linear model predicts more than 1 and less than zero, can't have these probs.
- 1 hour study, always has the same return.

What happens if we try to model these data with linear regression?

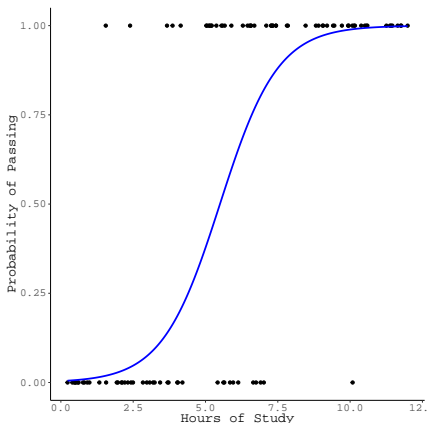
- Hmm...notice any problems?



# Logistic Regression Visualized

We get a much better model using logistic regression.

- The link function ensures legal predicted values.
- The sigmoidal curve implies fluctuation in the effectiveness of extra study time.
  - More study time is most beneficial for students with around 5.5 hours of study.



# Probabilities, Odds, & Odds-Ratios

In 2017, 2535 people participated in the *Ultra-Trail du Mont-Blanc*, but only 66.55% finished the race.

- Below, you can find a cross-tabulation of finishing status and sex.

Sex	Finish	
	No	Yes
Female	95	147
Male	753	1540

interpretation is in terms of odds!

Probability of success:  
success/total, in the  
discreet aspect.

$$P(\text{Fin}|F) = 147 / (147 + 95) \\ = \sim 0.607 = 60.7\%$$

- What is the *probability* of finishing for each sex?
- What are the *odds* of finishing for each sex?
- What is the *odds ratio* of finishing for males vs. females?

Odds of finishing  
 $O(\text{Fin}|F) = 147/95 \approx 1.55$

Odds ratio males vs  
females of finishing  
 $O(\text{Fin}|M)/O(\text{Fin}|F) =$   
 $2.045/1.547 = 1.32$

Use language of odds, not chance or probability.

# Defining the Logistic Regression Model

We are modelling a continuous thing

In logistic regression problems, we are modeling binary data:

- Usual coding:  $Y \in \{1 = \text{"Success"}, 0 = \text{"Failure"}\}$ .

The *Binomial* distribution is a good way to represent this kind of data.

- The systematic component in our logistic regression model will be the binomial distribution.

The mean of the binomial distribution (with  $N = 1$ ) is the "success" probability,  $\pi = P(Y = 1)$ .

- We are interested in modeling  $\mu_Y = \pi$ :

$$g(\pi) = \beta_0 + \sum_{p=1}^P \beta_p X_p$$



# Link Function for Logistic Regression

---

Because  $\pi$  is bounded by 0 and 1 and not linear related to  $\mathbf{X}$ , we cannot model it directly—we must apply an appropriate link function.

- Logistic regression uses the *logit link*.
- Given  $\pi$ , we can define the *odds* of success as:

$$O_s = \frac{\pi}{1 - \pi}$$

- Because  $\pi \in [0, 1]$ , we know that  $O_s \geq 0$ .
- We take the natural log of the odds as the last step to fully map  $\pi$  to the real line.

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





# Fully Specified Logistic Regression Model

---

Our final logistic regression model is:

$$Y \sim \text{Bin}(\pi, 1)$$
$$\text{logit}(\pi) = \beta_0 + \sum_{p=1}^P \beta_p X_p$$

The fitted model can be represented as:

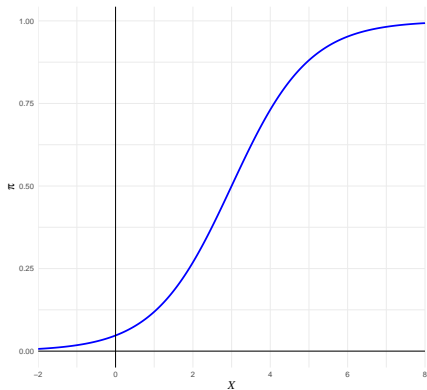
$$\text{logit}(\hat{\pi}) = \hat{\beta}_0 + \sum_{p=1}^P \hat{\beta}_p X_p$$

The fitted coefficients,  $\{\hat{\beta}_0, \hat{\beta}_p\}$ , are interpreted in units of *log odds*.

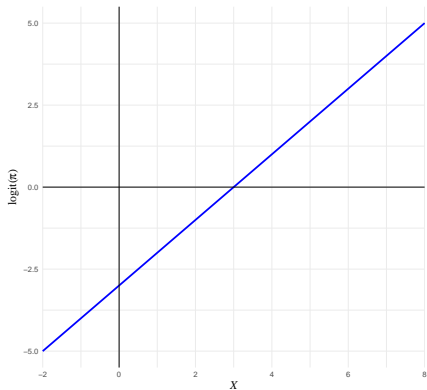


# Visualizing Logistic Regression

Probability Model



Logit Model



# Logistic Regression Example

---

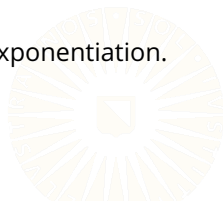
If we fit a logistic regression model to the test-passing data plotted above, we get:

$$\text{logit}(\hat{\pi}_{\text{pass}}) = -3.414 + 0.683X_{\text{study}}$$

- A student who does not study at all has -3.414 log odds of passing the exam.
- For each additional hour of study, a student's log odds of passing increase by 0.683 units.

Log odds do not lend themselves to interpretation.

- We can convert the effects back to an odds scale by exponentiation.
- $\hat{\beta}$  has log odds units, but  $e^{\hat{\beta}}$  has odds units.



# Interpretations

---

Problem: exponentiation non-linearizes data again. Slope must be interpreted as multiplicative instead of additive.

Exponentiating the coefficients also converts the additive effects to multiplicative effects.

- We can interpret  $\hat{\beta}$  as we would in linear regression:
  - A unit change in  $X_p$  produces an expected change of  $\hat{\beta}_p$  units in  $\text{logit}(\pi)$ .
- After exponentiation, however, unit changes in  $X_p$  imply multiplicative changes in  $O_s = \pi / (1 - \pi)$ .
  - A unit change in  $X_p$  results in multiplying  $O_s$  by  $e^{\hat{\beta}_p}$ .



# Interpretations

---

Exponentiating the coefficients in our toy test-passing example produces the following interpretations:

- A student who does not study is expected to pass the exam with odds of 0.033.
- For each additional hour a student studies, their odds of passing increase by 1.98 *times*.
  - Odds of passing are *multiplied* by 1.98 for each extra hour of study.



# Interpretations

---

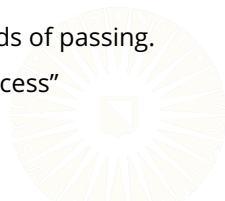
Exponentiating the coefficients in our toy test-passing example produces the following interpretations:

- A student who does not study is expected to pass the exam with odds of 0.033.
- For each additional hour a student studies, their odds of passing increase by 1.98 *times*.
  - Odds of passing are *multiplied* by 1.98 for each extra hour of study.

Due to the confusing interpretations of the coefficients, we often focus on the valance of the effects:

- Additional study time is associated with increased odds of passing.
- $\hat{\beta}_p > 0$  = "Increased Success",  $e^{\hat{\beta}_p} > 1$  = "Increased Success"

null point changes



# Example

---

Let's use logistic regression to compute the odds of finishing the UTMB.

```
## Read the UTMB data:  
utmb <- readRDS(paste0(dataDir, "utmb_finish_2017.rds"))
```

We use the `glm()` function to estimate generalized linear models.

- To get a logistic regression model, we need to do two things:
  1. Specify a binary outcome variable
  2. Specify the `family = "binomial"` argument.

```
## Estimate the logistic regression model:  
fit <- glm(Finish ~ Sex, family = binomial(link = "logit"), data = utmb)
```

logit is default

# Example

---

```
partSummary(fit, -1)

## Coefficients:
##           Estimate Std. Error z value Pr(>|z|)
## (Intercept)  0.4366     0.1316   3.316 0.000912
## SexMale      0.2789     0.1389   2.007 0.044712
##
## (Dispersion parameter for binomial family taken to be 1)
##
##      Null deviance: 3231.3  on 2534  degrees of freedom
## Residual deviance: 3227.3  on 2533  degrees of freedom
## AIC: 3231.3
##
## Number of Fisher Scoring iterations: 4
```



# Example

---

The raw coefficient estimates are in units of log-odds.

- We need to exponentiate the estimates to get odds ratios.

```
library(dplyr)
```

```
coef(fit) %>% exp()
```

```
## (Intercept)      SexMale  
##      1.547368      1.321697
```

Odds for female

odds RATIOS

See slide 22.

# Multiple Logistic Regression

---

The preceding example was a *simple logistic regression*.

- Including multiple predictor variables in the systematic component leads to *multiple logistic regression*.
- The relative differences between simple logistic regression and multiple logistic regression are the same as those between simple linear regression and multiple linear regression.
  - The only important complication is that the regression coefficients become partial effects.



# Example

---

Let's use logistic regression to predict the chances that Titanic passengers survived the sinking based on their age, sex, and ticket class.

```
## Read the data:  
titanic <- readRDS(paste0(dataDir, "titanic.rds"))  
  
## Estimate the logistic regression model:  
fit <- glm(survived ~ age + sex + class,  
           data = titanic,  
           family = "binomial")
```

# Example

```
partSummary(fit, -1)
```

Controlling for, partialling for the other variables.  
the log odds of males surviving is 2.5 lower than that of females.

```
## Coefficients:
```

```
##           Estimate Std. Error z value Pr(>|z|)
## (Intercept)  3.63492    0.37045   9.812 < 2e-16
## age         -0.03427    0.00716  -4.787 1.69e-06
## sexmale     -2.58872    0.18701 -13.843 < 2e-16
## class2nd    -1.19911    0.26158  -4.584 4.56e-06
## class3rd    -2.45544    0.25322  -9.697 < 2e-16
##
```

Intercept: for female, 1st  
class, yet to be born,  
survival odds is 3.6

```
## (Dispersion parameter for binomial family taken to be 1)
```

```
##
```

```
##      Null deviance: 1182.77  on 886  degrees of freedom
```

```
## Residual deviance:  801.59  on 882  degrees of freedom
```

```
## AIC: 811.59
```

```
##
```

```
## Number of Fisher Scoring iterations: 5
```

# Example

Always report the exponentiated and the non exponentiated value.

Compute odds ratios.

```
(or <- coef(fit) %>% exp())  
## (Intercept)      age      sexmale      class2nd      class3rd  
## 37.8988400  0.9663058  0.0751161  0.3014609  0.0858252
```

Odds ratios smaller than 1.0 can be difficult to explain.

- We can ease interpretation by reciprocating the estimates.

inverting the ratio

```
1 / or  
## (Intercept)      age      sexmale      class2nd      class3rd  
## 0.02638603  1.03486914 13.31272574  3.31717996 11.65158920
```

for every year older, odds ratio of survival is higher by 1.034

# Example

---

To convince ourselves that the above operation is sensible, we can compare the inverse odds ratios to the odds ratios we get from predicting the chances of dying.

```
library(magrittr)

fit2 <- titanic %>%
  mutate(died = relevel(survived, ref = "yes")) %$%
  glm(died ~ age + sex + class, family = "binomial")
```

# Example

---

```
partSummary(fit2, -1)

## Coefficients:
##           Estimate Std. Error z value Pr(>|z|)
## (Intercept) -3.63492    0.37045  -9.812  < 2e-16
## age         0.03427    0.00716   4.787 1.69e-06
## sexmale     2.58872    0.18701  13.843  < 2e-16
## class2nd    1.19911    0.26158   4.584 4.56e-06
## class3rd    2.45544    0.25322   9.697  < 2e-16
##
## (Dispersion parameter for binomial family taken to be 1)
##
##      Null deviance: 1182.77  on 886  degrees of freedom
## Residual deviance:  801.59  on 882  degrees of freedom
## AIC: 811.59
##
## Number of Fisher Scoring iterations: 5
```

# Example

---

We get the same odds ratios that we derived through reciprocation.

```
coef(fit2) %>% exp()

## (Intercept)          age      sexmale    class2nd    class3rd
## 0.02638603  1.03486914 13.31272574  3.31717996 11.65158920

1 / or

## (Intercept)          age      sexmale    class2nd    class3rd
## 0.02638603  1.03486914 13.31272574  3.31717996 11.65158920
```



# Example in Equations

---

Here's the symbolic representation of our logistic regression model:

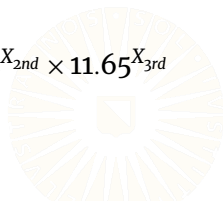
$$\text{logit}(\pi_{died}) = \beta_0 + \beta_1 X_{age} + \beta_2 X_{male} + \beta_3 X_{2nd} + \beta_4 X_{3rd}$$

By fitting this model to the *titanic* data we get:

$$\text{logit}(\hat{\pi}_{died}) = -3.63 + 0.03 X_{age} + 2.59 X_{male} + 1.2 X_{2nd} + 2.46 X_{3rd}$$

Exponentiating the coefficients produces:

$$\frac{\hat{\pi}_{died}}{1 - \hat{\pi}_{died}} = \frac{\hat{\pi}_{died}}{\hat{\pi}_{survived}} = 0.03 \times 1.03^{X_{age}} \times 13.31^{X_{male}} \times 3.32^{X_{2nd}} \times 11.65^{X_{3rd}}$$



# Exponentiating the Systematic Component

$$\text{logit}(\hat{\pi}_{died}) = -3.63 + 0.03X_{age} + 2.59X_{male} + 1.2X_{2nd} + 2.46X_{3rd}$$

$$e^{\text{logit}(\hat{\pi}_{died})} = e^{(-3.63 + 0.03X_{age} + 2.59X_{male} + 1.2X_{2nd} + 2.46X_{3rd})}$$

$$\begin{aligned}\frac{\hat{\pi}_{died}}{\hat{\pi}_{survived}} &= e^{-3.63} \times e^{0.03X_{age}} \times e^{2.59X_{male}} \times e^{1.2X_{2nd}} \times e^{2.46X_{3rd}} \\ &= e^{-3.63} \times (e^{0.03})^{X_{age}} \times (e^{2.59})^{X_{male}} \times (e^{1.2})^{X_{2nd}} \times (e^{2.46})^{X_{3rd}} \\ &= 0.03 \times 1.03^{X_{age}} \times 13.31^{X_{male}} \times 3.32^{X_{2nd}} \times 11.65^{X_{3rd}}\end{aligned}$$

# Model Comparison

---

```
## Estimate a restricted model:
fit0 <- update(fit, ". ~ . - class")

## Check the result:
partSummary(fit0, 1:3)

## Call:
## glm(formula = survived ~ age + sex, family = "binomial", data = titanic)
##
## Coefficients:
##              Estimate Std. Error z value Pr(>|z|)
## (Intercept)  1.113881   0.208401   5.345 9.05e-08
## age         -0.002060   0.005865  -0.351  0.725
## sexmale     -2.500001   0.167772 -14.901 < 2e-16
##
## (Dispersion parameter for binomial family taken to be 1)
```

# Model Comparison

---

We don't have an  $R^2$  statistic for logistic regression models, so we need to use a *likelihood ratio test* to compare nested models.

```
anova(fit0, fit, test = "LRT")

## Analysis of Deviance Table
##
## Model 1: survived ~ age + sex
## Model 2: survived ~ age + sex + class
##   Resid. Df Resid. Dev Df Deviance  Pr(>Chi)
## 1         884       916.00
## 2         882       801.59  2   114.41 < 2.2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

# Model Comparison

---

We can also use information criteria.

```
AIC(fit0, fit)
```

```
##          df          AIC
## fit0     3 921.9989
## fit      5 811.5940
```

```
BIC(fit0, fit)
```

```
##          df          BIC
## fit0     3 936.3624
## fit      5 835.5333
```

# CLASSIfication



# Predictions from Logistic Regression

---

Given a fitted logistic regression model, we can get predictions for new observations of  $\{X_p\}$ ,  $\{X'_p\}$ .

- Directly applying  $\{\hat{\beta}_0, \hat{\beta}_p\}$  to  $\{X'_p\}$  will produce predictions on the scale of  $\eta$ :

$$\hat{\eta}' = \hat{\beta}_0 + \sum_{p=1}^P \hat{\beta}_p X'_p$$

- By applying the inverse link function,  $g^{-1}(\cdot)$ , to  $\hat{\eta}'$ , we get predicted success probabilities:

$$\hat{\pi}' = g^{-1}(\hat{\eta}')$$



# Predictions from Logistic Regression

---

In logistic regression, the inverse link function,  $g^{-1}(\cdot)$ , is the *logistic function*:

$$\text{logistic}(X) = \frac{e^X}{1 + e^X}$$

So, we convert  $\hat{\eta}'$  to  $\hat{\pi}'$  by:

$$\hat{\pi}' = \frac{e^{\hat{\eta}'}}{1 + e^{\hat{\eta}'}} = \frac{\exp\left(\hat{\beta}_0 + \sum_{p=1}^P \hat{\beta}_p X'_p\right)}{1 + \exp\left(\hat{\beta}_0 + \sum_{p=1}^P \hat{\beta}_p X'_p\right)}$$





# Classification with Logistic Regression

---

Once we have computed the predicted success probabilities,  $\hat{\pi}'$ , we can use them to classify new observations.

- By choosing a threshold on  $\hat{\pi}'$ , say  $\hat{\pi}' = t$ , we can classify the new observations as “Successes” or “Failures”:

$$\hat{Y}' = \begin{cases} 1 & \text{if } \hat{\pi}' \geq t \\ 0 & \text{if } \hat{\pi}' < t \end{cases}$$



# Classification Example

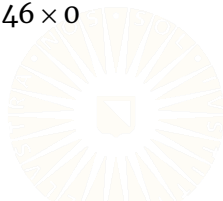
---

Say we want to classify a hypothetical passenger as either having died or survived the sinking.

- Assume this passenger has the following characteristics:
  - They are 17 years old
  - They are male
  - They are a second class passenger

First we plug their predictor data into the fitted model to get their model-implied  $\eta$ :

$$\begin{aligned}\hat{\eta}_{died} &= -3.63 + 0.03 \times 17 + 2.59 \times 1 + 1.2 \times 1 + 2.46 \times 0 \\ &= 0.736\end{aligned}$$



# Classification Example

---

Next we convert the predicted  $\eta$  value into a model-implied success probability by applying the logistic function:

$$\frac{e^{0.736}}{1 + e^{0.736}} = 0.676$$

Finally, to make the classification, assume a threshold of  $\hat{\pi}' = 0.5$  as the decision boundary.

- Because  $0.676 > 0.5$  we would classify this passenger as having died.

