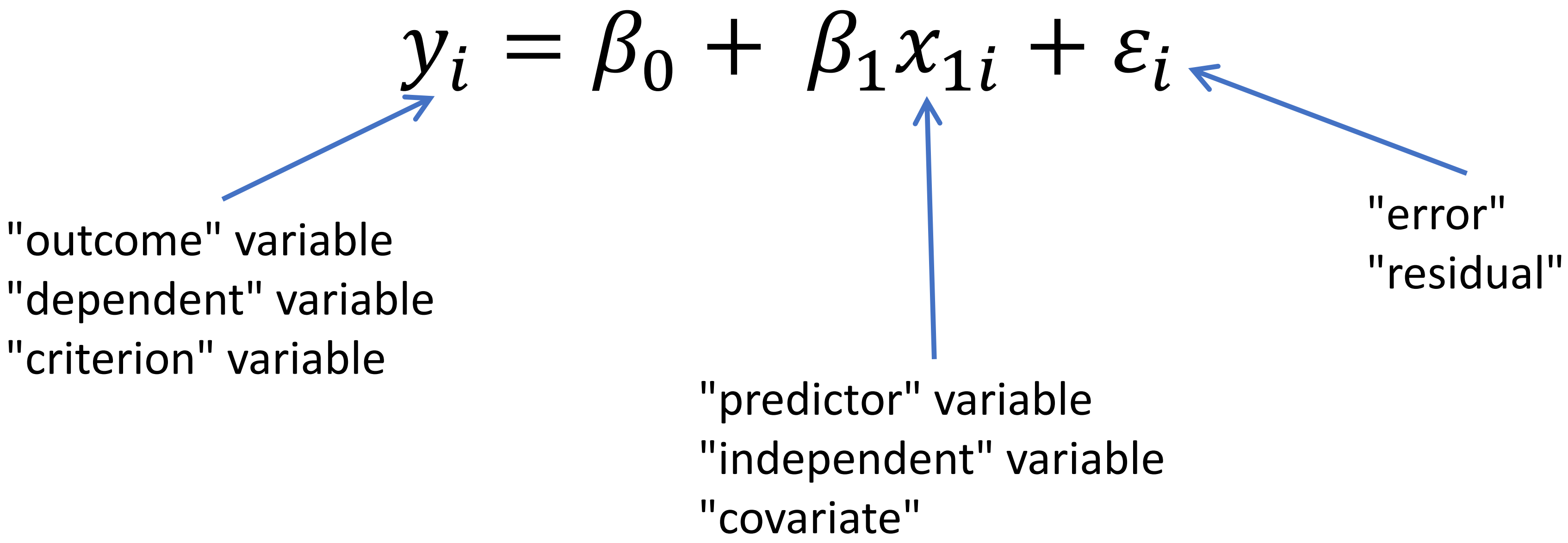
Generalized Linear Models

February 9, 2022

Regression

$$y_i = \beta_0 + \ \beta_1 x_{1i} + \varepsilon_i$$

Regression



$$y_i = \beta_0 + \beta_1 x_{1i} + \varepsilon_i$$

"outcome" variable
"dependent" variable
"criterion" variable

"predictor" variable
"independent" variable
"covariate"

"error"
"residual"

- assume that relationship is linear and observations are independent
- assume that residuals are distributed normally with mean 0 and constant variance
- variability of error does not depend on value of x (assumption of homoscedasticity)

Regression assumptions

- Model is built for a **continuous** outcome variable with normally distributed residuals

- What if outcome is **discrete**?
  - pain ("how much pain are you feeling?") on a 1-10 scale
  - (you may not want to assume that 5 is 5 times more pain than 1)

  - outcome of an individual trial is a success (1) or a failure (0)

Discrete outcomes

- Binary 0/1 outcome: nothing will stop you from running `lm` or `lmer`

- Expected values are really predicted probabilities –
  - could fall outside [0, 1] range
  - normality of residuals violated
  - violation of homoscedasticity (variance gets compressed at ends of range)
  - -> biased parameter estimates and incorrect p-values!

What is the solution?

- expand our concept of regression model:
- linear predictor: optimal linear combination of predictor variables x_i
- response distribution: probability distribution for outcome variable
- link function: relate the linear predictor to the response distribution

- for standard linear regression, the link function is just an identity: the optimal linear combination predicts the outcome variable (y) as a normal distribution with mean $\hat{y}_i$ and constant variance – just add up predictors and multiply by appropriate coefficients

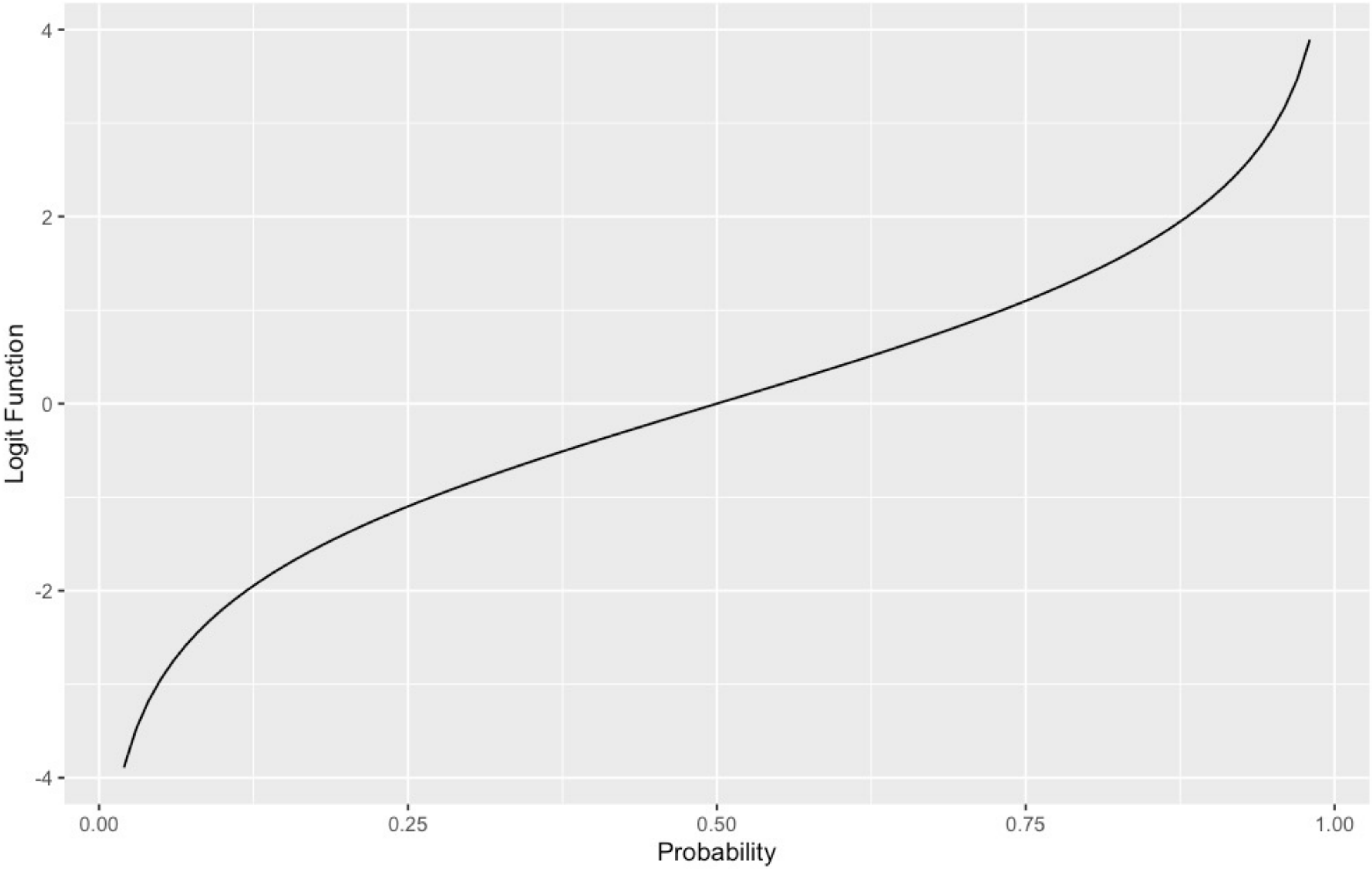
Logistic (binomial) regression

- If our outcome can only be between 0 and 1 inclusive, we cannot just use an identity link function because there is nothing to constrain the prediction (could predict values < 0 or > 1)

- Instead, use *logit* function that transforms outcome bounded by 0 and 1 to continuous range

$$logit(p) = \ln(\frac{p}{1\ - p})$$

- Now, your linear predictor predicts <u>logit(*p*)</u>


4
2
0
-2
-4
Logit Function
0.00
0.25
0.50
0.75
1.00
Probability

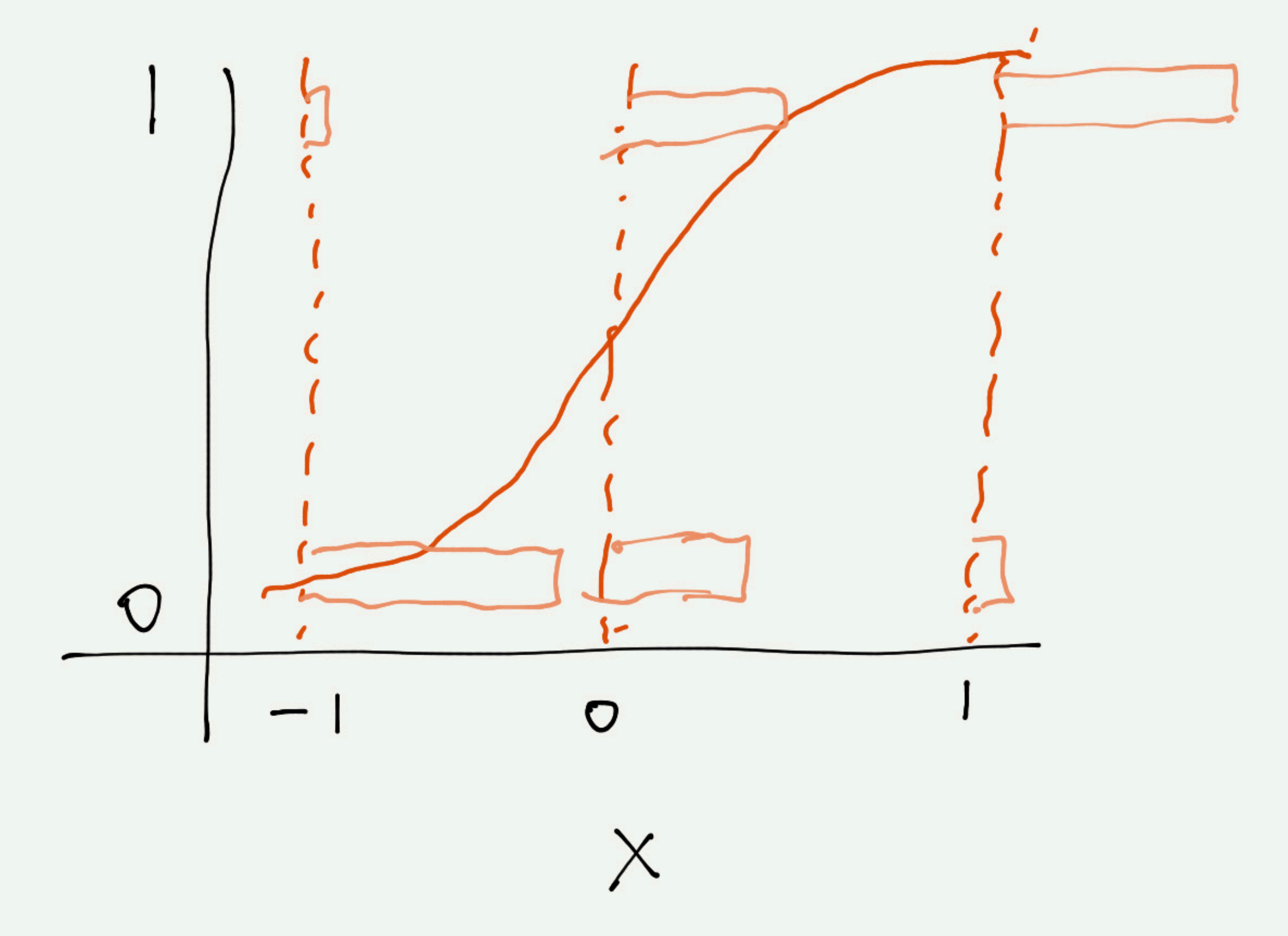
	Linear Regression	Binomial (Logistic) Regression
Linear predictor	$\eta_i = \beta_0 + \beta_1 x_{1i} + \cdots$	
Link function		
Response distribution	$y_i \vert \mu_i \sim N(\mu_i, \sigma^2)$	

	Linear Regression	Binomial (Logistic) Regression
Linear predictor	$\eta_i = \beta_0 + \beta_1 x_{1i} + \cdots$	
Link function	$\mu_i = \eta_i$	
Response distribution	$y_i \mid \mu_i \sim N(\mu_i, \sigma^2)$	

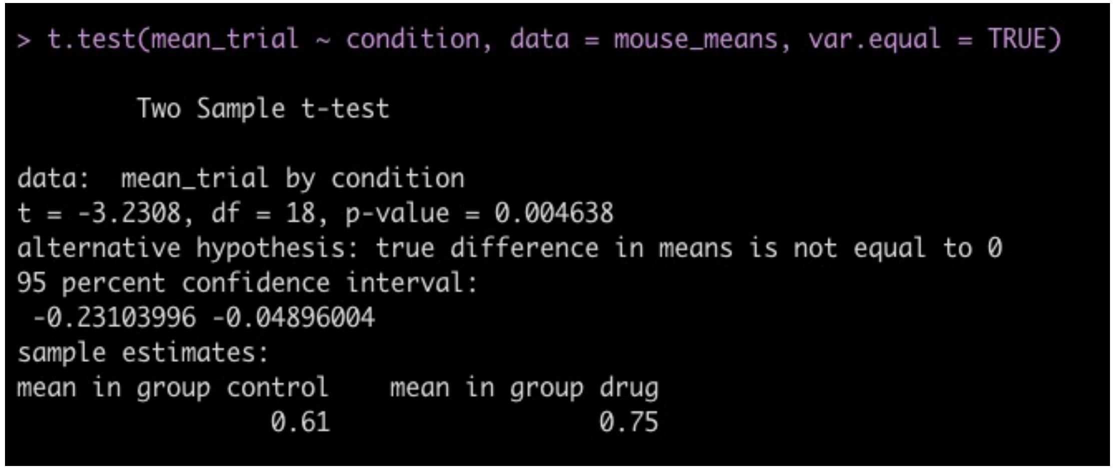
	Linear Regression	Binomial (Logistic) Regression
Linear predictor	$\eta_i = \beta_0 + \beta_1 x_{1i} + \cdots$	$\eta_i = \beta_0 + \beta_1 x_{1i} + \cdots$
Link function	$\mu_i = \eta_i$	
Response distribution	$y_i \mid \mu_i \sim N(\mu_i, \sigma^2)$	

	Linear Regression	Binomial (Logistic) Regression
Linear predictor	$\eta_i = \beta_0 + \beta_1 x_{1i} + \cdots$	$\eta_i = \beta_0 + \beta_1 x_{1i} + \cdots$
Link function	$\mu_i = \eta_i$	$\mathrm{logit}(\mu_i) = \eta_i$
Response distribution	$y_i \| \mu_i \sim N(\mu_i, \sigma^2)$	$y_i \| \mu_i \sim \mathrm{Bernouilli}(\mu_i)$

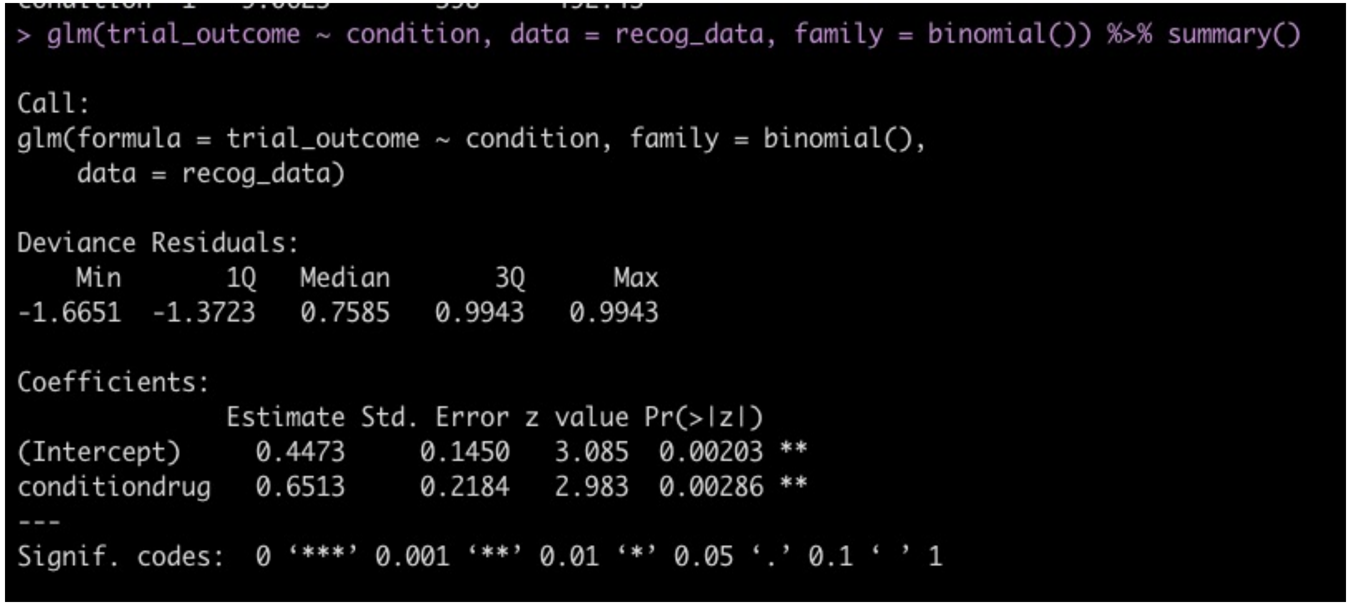
	Linear Regression	Binomial (Logistic) Regression
Linear predictor	$\eta_i = \beta_0 + \beta_1 x_{1i} + \cdots$	$\eta_i = \beta_0 + \beta_1 x_{1i} + \cdots$
Link function	$\mu_i = \eta_i$	$\text{logit}(\mu_i) = \eta_i$
Response distribution	$y_i \vert \mu_i \sim N(\mu_i, \sigma^2)$	$y_i \vert \mu_i \sim \text{Bernouilli}(\mu_i)$



Use GLM to analyze trials rather than means



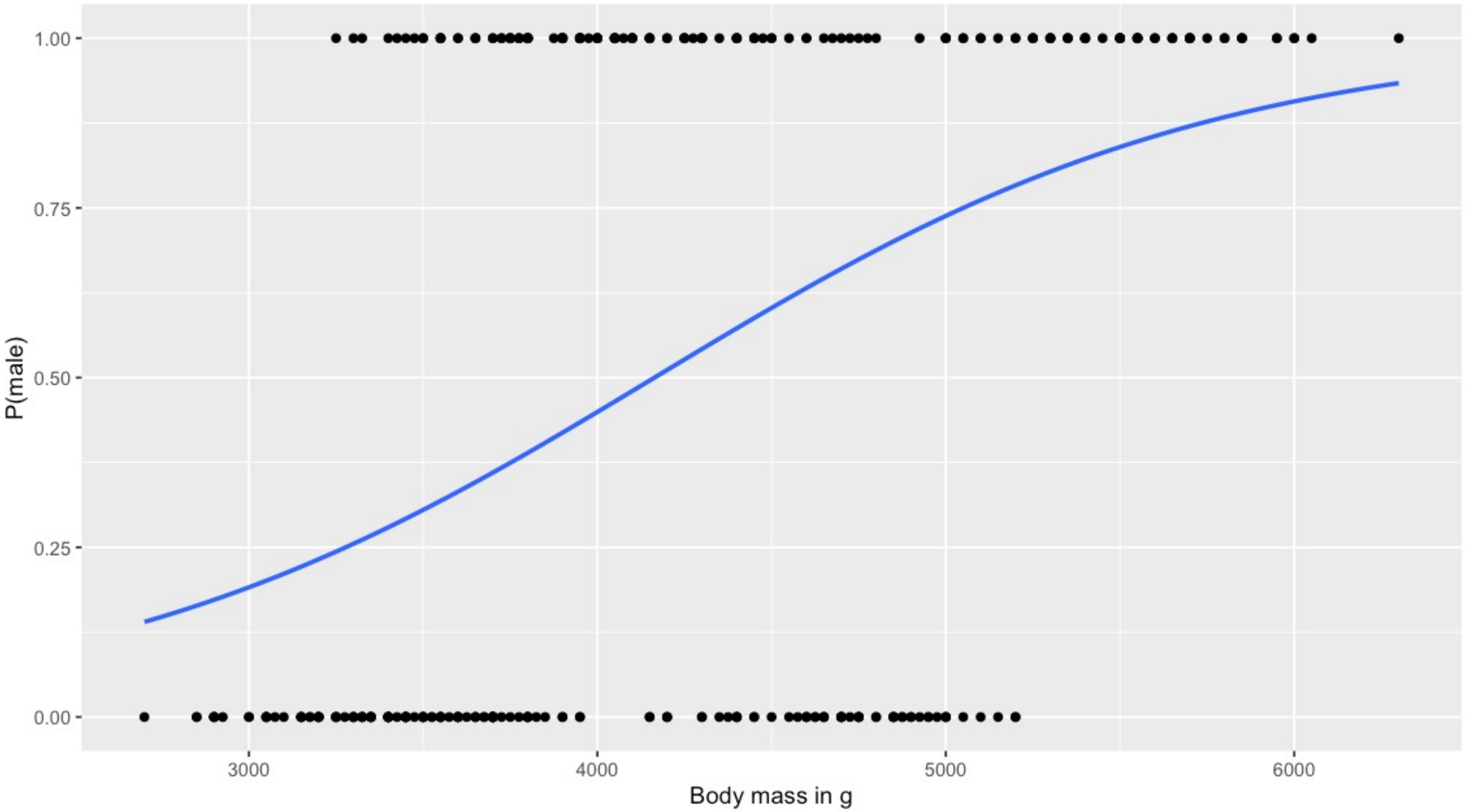
```
> t.test(mean_trial ~ condition, data = mouse_means, var.equal = TRUE)

        Two Sample t-test

data:  mean_trial by condition
t = -3.2308, df = 18, p-value = 0.004638
alternative hypothesis: true difference in means is not equal to 0
95 percent confidence interval:
 -0.23103996 -0.04896004
sample estimates:
mean in group control    mean in group drug
                 0.61                  0.75
```



```
> glm(trial_outcome ~ condition, data = recog_data, family = binomial()) %>% summary()

Call:
glm(formula = trial_outcome ~ condition, family = binomial(),
    data = recog_data)

Deviance Residuals:
    Min       1Q   Median       3Q      Max
-1.6651  -1.3723   0.7585   0.9943   0.9943

Coefficients:
              Estimate Std. Error z value Pr(>|z|)
(Intercept)     0.4473     0.1450   3.085  0.00203 **
conditiondrug   0.6513     0.2184   2.983  0.00286 **
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1
```

20220209_logistic.R

Penguins!



20220209_logistic.R

Many link functions

- Binomial/logistic
  - Logistic (0,1) – special case
  - Binomial will accept cbind(successes, failures) as a predictor
  - If you have successes and total_trials:
    - glmer(cbind(successes,total_trials-successes) ~ drug*delay + (1|monkey), family = binomial(), data = my_data) will work

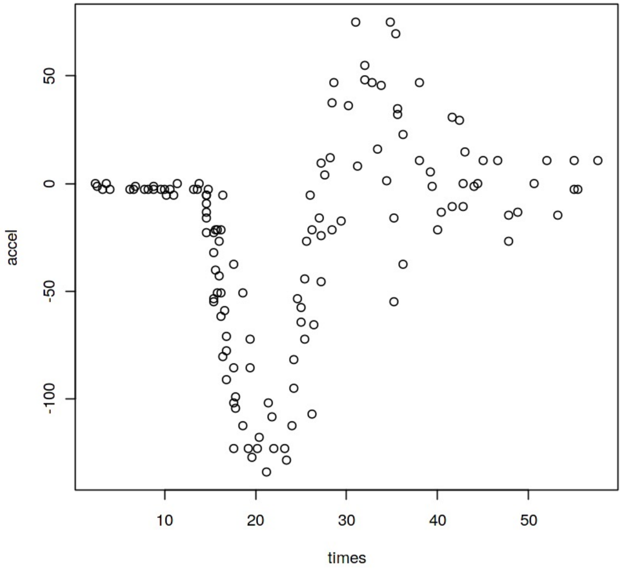
Many link functions

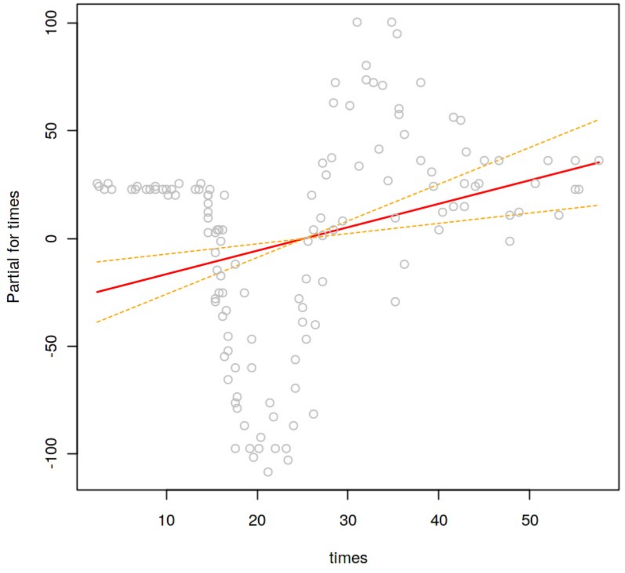
- Poisson
  - Modeling count data "rare events model"
  - May fit data better when you have small number of discrete outcomes (kinds of synapses, number of gold particles)
  - family = poisson() in glm/glmer

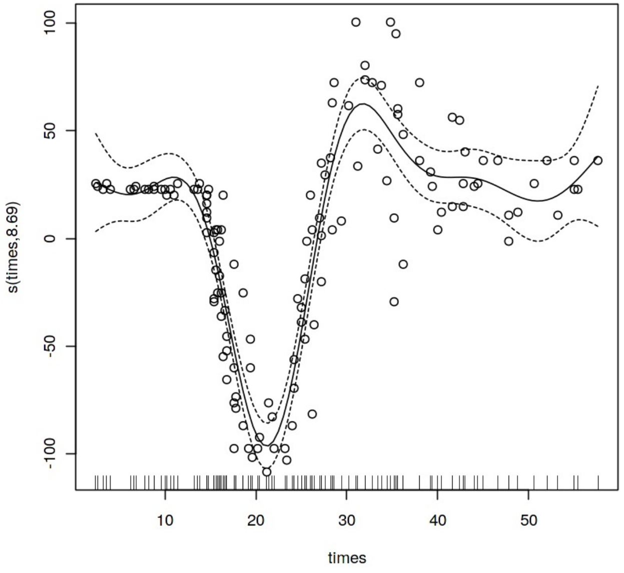
- Quasibinomial / quasi-Poisson
  - Includes additional parameter for unexplained variance
  - Binomial and Poisson variance is function of mean
  - Underdispersion / overdispersion
  - Zero-inflation

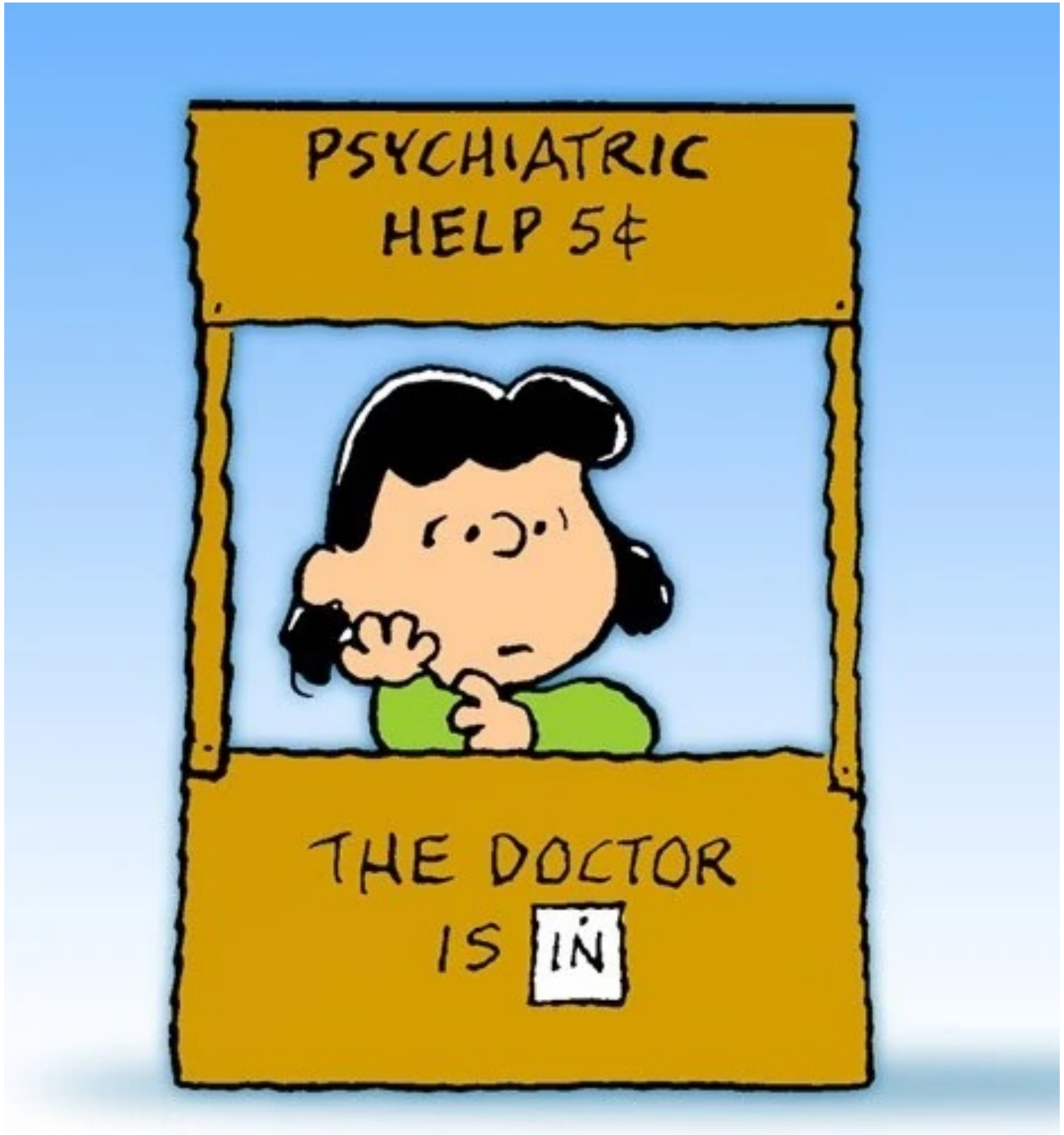
Generalized additive models (GAMs)

- uses `mgcv` package in R
- does not assume linearity of relationship between predictors and outcome: nature of function unknown/arbitrary
- https://noamross.github.io/gams-in-r-course/








PSYCHIATRIC
HELP 5¢
THE DOCTOR
IS IN