EPSE 596: Correlational Designs & Analysis

Ed Kroc

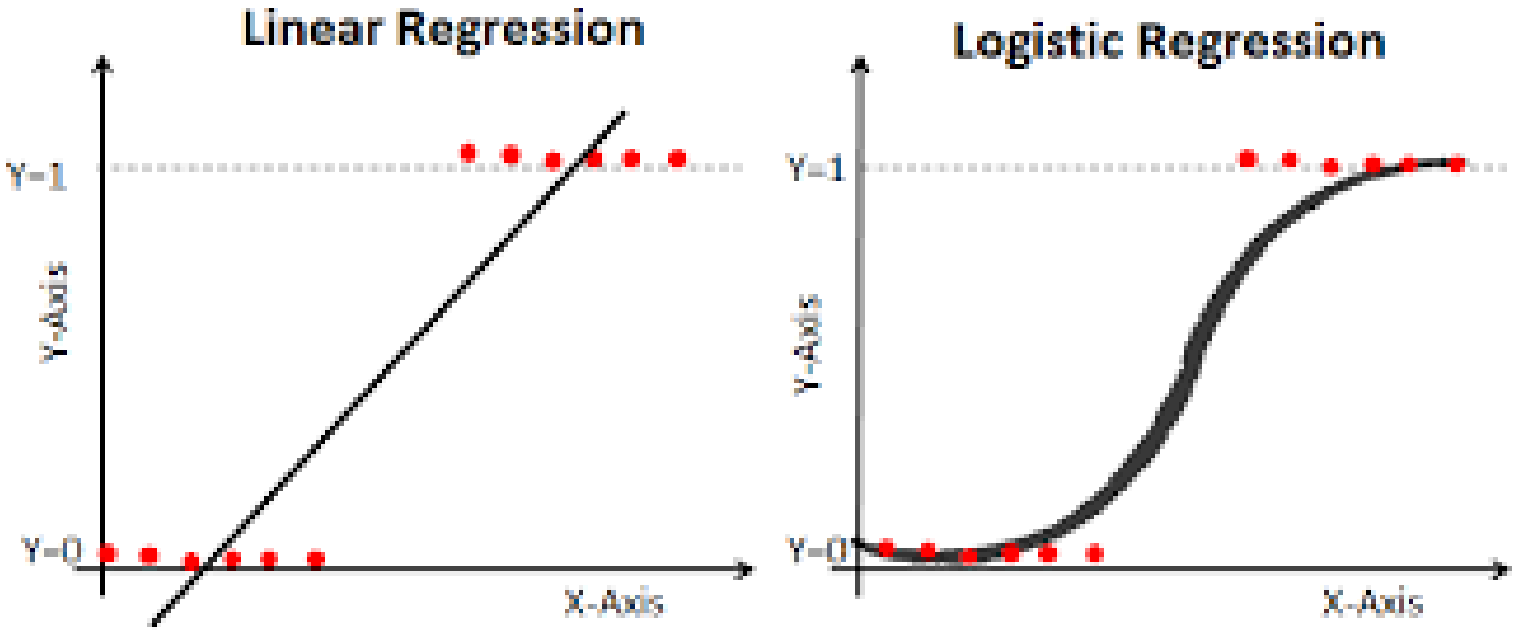
University of British Columbia

ed.kroc@ubc.ca

Binary response data

- In this class, we have so far only studied *ordinary linear regression models* which are (potentially) applicable to situations where our response data are *continuous*.
- However, our techniques can be generalized to handle more diverse response phenomena via *generalized linear models (GLMs)*, which can then be generalized further via *generalized linear mixed models (GLMMs)*.
- GLMs adapt the ordinary linear regression framework to a variety of settings where our response data are not continuous and/or where our model errors should not be assumed to be normally distributed.
- Most common non-continuous/non-normal situation: *binary 0/1 response data*, giving rise to *binomial regression*. Very brief intro this week.
- First half of EPSE 682 (Jan-Apr 2022) deals with more GLMs; future iterations of EPSE 683 (Jan-Apr 2023) will deal with GLMMs.

Binomial regression for Binomial response data



- If your response data are *binary*, then using ordinary linear regression is going to be a bad idea: you will get predicted/fitted values that make no sense!
- Moreover, when Y is binary, it is really only $\Pr(Y = 1)$ that we care about (see next slides). But probabilities have to live on $[0, 1]$.

Binomial regression for Binomial response data

- If a random quantity Y is binary, then it can always be encoded as 0 or 1. This is called a *Bernoulli* random variable (a special case of a *Binomial* random variable), and is totally parameterized/characterized by knowing $p = \Pr(Y = 1)$. We write $Y \sim Ber(p)$.
- Notice that $p \in [0, 1]$.
- Notice that p completely characterizes the distribution of Y; e.g.

$$\Pr(Y = 0) = 1 - \Pr(Y = 1) = 1 - p.$$

$$\mathbb{E}(Y) = p, \qquad \text{Var}(Y) = p(1 - p).$$

- This is in stark contrast to a *normal* random variable, where there are two free parameters: $\mu \in \mathbb{R}$ that determines the mean, and $\sigma \in \mathbb{R}^+$ that (independently) determines the variance.

Binomial regression for Binomial response data

- So $p = \Pr(Y = 1)$ totally captures the distribution of the response Y.
- Moreover, this probability equals the *mean* of the response Y.
- So could try to use our same ordinary linear regression machinery as before to model the mean, e.g.,

$$\Pr(Y = 1) = \beta_0 + \beta_1 X$$

- But in ordinary linear regression there were no restrictions on the lefthand side.
- Now, however, $0 \leqslant \Pr(Y = 1) \leqslant 1$.
- Trying to do ordinary least squares or maximum likelihood (or other estimation) approaches to the regression equation while simultaneously enforcing the added restrictions is a mathematical nightmare. **Solution:** Transform the restrictions away.

Binomial regression for Binomial response data

- There are many different functions that will transform probabilities (#'s between 0 and 1) into numbers that span all of $\mathbb{R}$. The most commonly used are:
  - **Logit or inverse logistic function:**

$$g_1(\pi) = \log\left(\frac{\pi}{1-\pi}\right)$$

  - Probit or inverse Normal function:

$$g_2(\pi) = \Phi^{-1}(\pi), \quad \text{(inverse CDF of standard Normal)}$$

  - Complementary log-log function:

$$g_3(\pi) = \log(-\log(1-\pi))$$

  - Log-log function:

$$g_4(\pi) = -\log(-\log(\pi))$$

Logistic regression for Binomial response data

- The standard *logistic regression* model for binomial response data (on one predictor X) is:

$$\log\left(\frac{\pi}{1-\pi}\right) = \beta_0 + \beta_1 X$$

  where $\pi := \Pr(Y = 1)$.

- We can solve this explicitly for the "success" probability π:

$$\pi = \frac{\exp(\beta_0 + \beta_1 X)}{1 + \exp(\beta_0 + \beta_1 X)} = \frac{1}{1 + \exp(-(\beta_0 + \beta_1 X))}$$

- This gives us an explicit expression for $\Pr(Y = 1)$.

- Note: a function of the form $\frac{\exp(x)}{1+\exp(x)}$ is called a *logistic* or *sigmoid function*.

Logistic regression for Binomial response data

- The Bernoulli/Binomial response data generate a likelihood function for this model:

$$\ell(\pi \mid (x_1, y_1), \ldots, (x_n, y_n)) = \prod_{i=1}^{n} \binom{m_i}{y_i} \pi^{y_i} (1-\pi)^{m_i - y_i},$$

  where we plug-in our proposed regression model (previous slide) for the key unknown parameter π.

- One can use this function then to derive the MLEs of the model coefficients, the β_j's, just as in ordinary linear regression.

- In general though, the MLEs won't have nice closed form expressions like we had in linear regression, but they can be very well approximated by a variety of numerical techniques (most notably, *iterative weighted least squared*).

- Note too: the ordinary least squares (OLS) estimation approach will not work well here; this is the first instance where OLS $\neq$ MLE.

Logistic regression: downsides

- Recall: In Week 8, I advised against transforming your raw data for a variety of reasons.
- Yet, logistic regression transforms the response data (what we are modelling) to make the mathematics tractable. Was necessary to remove the extra restrictions we would have had to impose on the regression model (i.e., $0 \leqslant p \leqslant 1$).
- But the transformation comes at a cost:
  - Interpreting the coefficients in our regression model is now more challenging and less intuitive; requires thinking about *odds of success* and *odds ratios* rather than just local/global linear effects (slopes).
  - Statistical power is greatly affected. If you generally want at least 10 data points per unknown parameter in an ordinary linear regression model, then you want about 80-100 data points per unknown parameter in a logistic regression model to achieve comparable power.
- These tradeoffs are necessary evils if we want to build reasonable regression models for binary response data.

Logistic regression: odds and odds ratios

- Standard logistic regression model for binary response data:

$$\log\left(\frac{\pi}{1-\pi}\right) = \beta_0 + \beta_1 X$$

- Simply exponentiating the standard logistic regression model, we find:

$$\frac{\pi}{1-\pi} = e^{\beta_0+\beta_1 X}$$

- The lefthand side of the equation is just the *odds* of observing $Y = 1$ (for a fixed but arbitrary value of X):

$$odds(Y = 1) = \frac{\Pr(Y = 1)}{\Pr(Y = 0)}$$

- Odds are *not* the same thing as probability (they can be any non-negative value), but encode similar information. For example:
  - The probability of flipping heads on a fair coin is $\frac{1}{2}$.
  - The odds of flipping heads on a fair coin are $\frac{1/2}{1/2} = 1$, or we often say are "1 to 1."

Logistic regression: odds and odds ratios

- Odds are *not* the same thing as probability (they can be any non-negative value), but encode similar information (like a "likelihood"). For example:
  - The probability of drawing an ace from a standard deck of 52 cards is $\frac{4}{52} = \frac{1}{13}$.
  - The odds of drawing an ace from a standard deck of 52 cards are $\frac{4/52}{48/52} = \frac{1}{12}$, or we often say are "1 to 12."
- Generally speaking, $odds(Y = 1) \approx \Pr(Y = 1)$ only when $\Pr(Y = 1)$ is *small*.
- This fact has clinical importance sometimes, e.g. when modelling the incidence rate of a rare disease. Then, odds can be treated as a probability (more intuitive).

Logistic regression: odds and odds ratios

- Back to the simple logistic regression model for binary Y on one predictor:

$$\frac{\pi}{1-\pi} = odds(Y=1) = e^{\beta_0+\beta_1 X}$$

- To interpret (average) rates of change in the transformed response (probability of success), one considers an *odds ratio*:

$$\frac{odds(Y=1 \mid X=x)}{odds(Y=1 \mid X=x-1)} = \frac{e^{\beta_0+\beta_1 x}}{e^{\beta_0+\beta_1(x-1)}} = e^{\beta_1}$$

- Thus, the quantity e^{β_1} tells us how much we would expect the odds of conditional "success" to change *multiplicatively* when X changes by 1 unit.

- For instance, if $\beta_1 = 0$ (corresponding to no estimated linear effect of X on the logit of Y), then the odds ratio equals 1; i.e. changing X doesn't seem to change the expected odds of success in Y at all.

Logistic regression: odds and odds ratios

- Consider the *odds ratio*:

$$\frac{odds(Y = 1 \mid X = x)}{odds(Y = 1 \mid X = x - 1)} = \frac{e^{\beta_0 + \beta_1 x}}{e^{\beta_0 + \beta_1 (x-1)}} = e^{\beta_1}$$

- Thus, the quantity e^{β_1} tells us how much we would expect the odds of conditional "success" to change *multiplicatively* when X changes by 1 unit.
- If β_1 is a large positive number, say $\beta_1 = 10$ (corresponding to a large, positive estimated linear effect of X on the logit of Y), then the odds ratio equals about 22,000; i.e. increasing X by a single unit greatly increases the expected odds of success in Y.
- Simiarly, if β_1 is a large negative number, say $\beta_1 = -8$ (corresponding to a large, negative estimated linear effect of X on the logit of Y), then the odds ratio equals about 0.00033; i.e. increasing X by a single unit greatly decreases the expected odds of success in Y.

Logistic regression: odds and odds ratios

- Equivalently, one can talk about the *log-odds ratio* rather than the ordinary odds ratio:

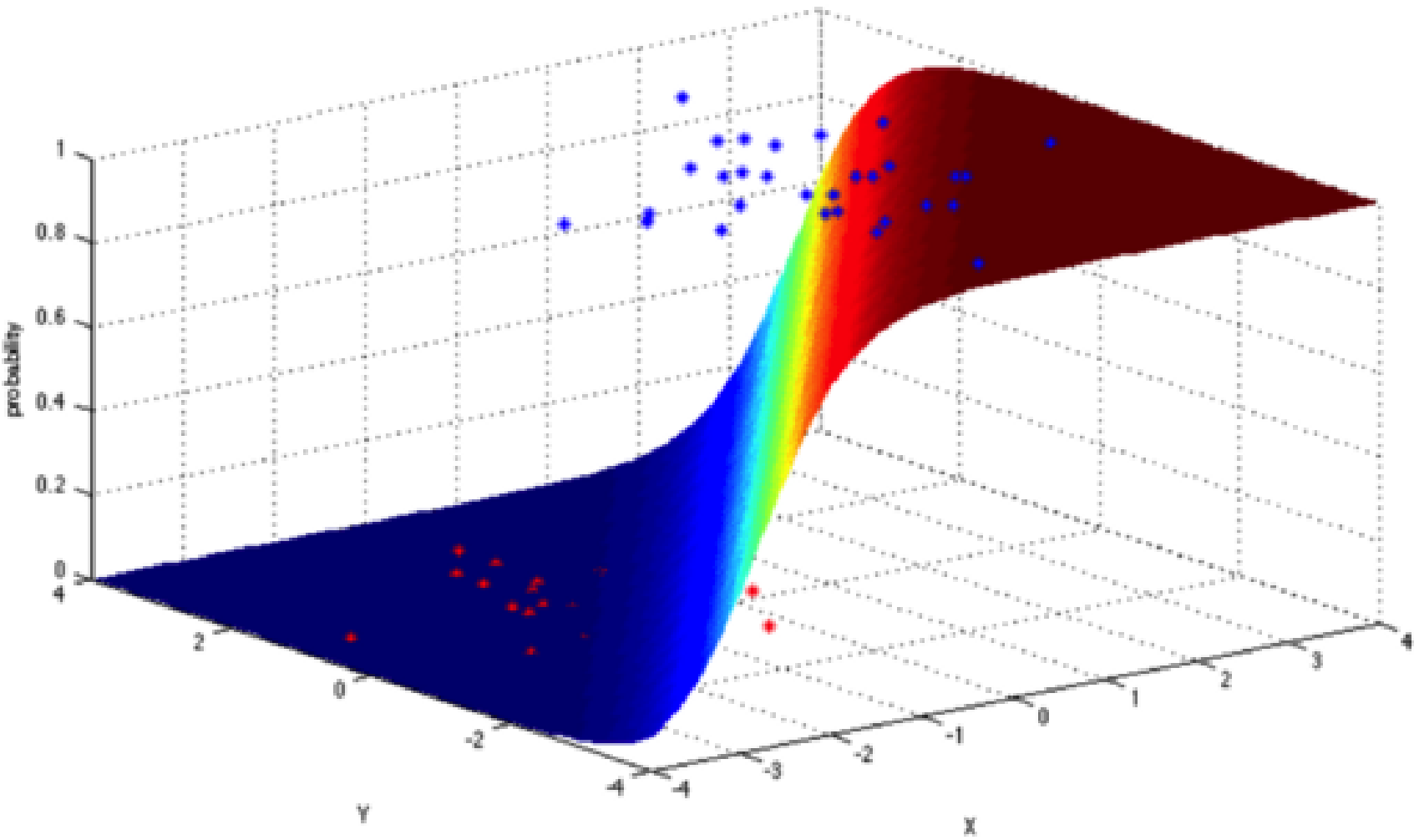
$$\begin{aligned}\beta_1 &= \log\left[\frac{odds(Y=1 \mid X=x)}{odds(Y=1 \mid X=x-1)}\right] \\ &= \log[odds(Y=1 \mid X=x)] - \log[odds(Y=1 \mid X=x-1)]\end{aligned}$$

- The analogous interpretations hold (since the logarithm is a monotone increasing function), but now we speak about increases or decreases in log-odds rather than just odds.

- You will see different people use either scale to interpret their model coefficients. Regardless, it is always about describing if a change in a predictor X corresponds to an increase or decrease in the odds/probability of a success in Y. [Notice: the odds can increase if and only if the probability of success increases.]

Logistic regression

- Note: all this readily extends to multiple predictors.



Logistic regression

- Logistic regression can also be extended to handle *multinomial* response data, e.g. $Y = 0$, 1, or 2, and with either ordered or non-ordered response categories.
- In general, **avoid these types of multinomial logistic regression models at all costs**, unless you have tons of data (where machine learning, etc. methods may be appropriate).
- Multinomial logistic regression suffers from even greater lack of interpretability and even less power.
- Additionally, multinomial regression requires making further modelling assumptions that are often totally unreasonable in practice.
- Binomial regression is hard enough; stick with it.
- My advice: Perform multiple Binomial logistic regressions if you have truly multinomial data.

Logistic regression: An example

- Here, we have 100 simulated observations on a binary response variable Y, and three predictors X_1, X_2, and X_3:

y	x1	x2	x3
0	-0.533268729	0.69435006	5.041927
1	1.214882517	0.07593485	5.332592
0	0.752889395	0.61961465	3.495342
0	0.215850088	0.67831539	3.096726
0	0.888656753	0.28373163	3.138944
1	0.788468455	0.19344354	3.133692
0	0.143666538	0.47781086	5.159349
1	-0.223556873	0.14079549	3.636836
0	0.870797261	1.05458425	4.399166
0	0.392492768	0.24044534	4.343689

- Start off by proposing a first-order model:

$$\text{logit}(\Pr(Y=1)) = \log\left(\frac{\Pr(Y=1)}{1-\Pr(Y=1)}\right) = \beta_0 + \beta_1 X_1 + \beta_2 X_2 + \beta_3 X_3$$

Logistic regression: An example

- Regular MLE-based fit (using IWLS) in R yields:

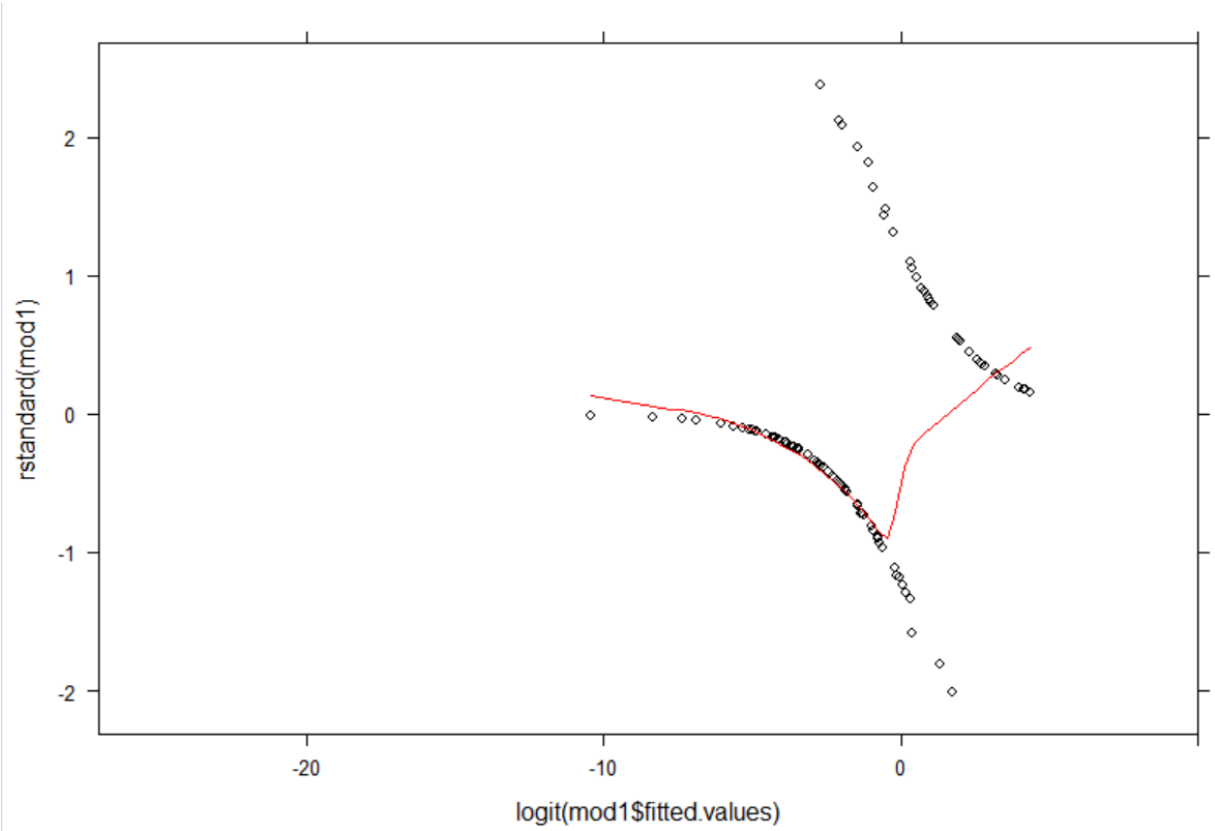
```
Call:
glm(formula = y ~ x1 + x2 + x3, family = binomial(link = "logit"))

Deviance Residuals:
    Min       1Q   Median       3Q      Max
-1.9488  -0.5017  -0.1695   0.3545   2.3401

Coefficients:
            Estimate Std. Error z value Pr(>|z|)
(Intercept) -0.01881    1.63639  -0.011 0.990830
x1          -2.50768    0.52000  -4.822 1.42e-06 ***
x2          -6.06491    1.72511  -3.516 0.000439 ***
x3           0.38791    0.39079   0.993 0.320892
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

(Dispersion parameter for binomial family taken to be 1)

    Null deviance: 128.207  on 99  degrees of freedom
Residual deviance:  66.756  on 96  degrees of freedom
AIC: 74.756

Number of Fisher Scoring iterations: 6
```

- Model estimates are on the raw *log-odds* scale.
- *Residual deviance* takes the place of *residual standard error*; smaller is better (i.e., more variation explained by model).

Logistic regression: An example

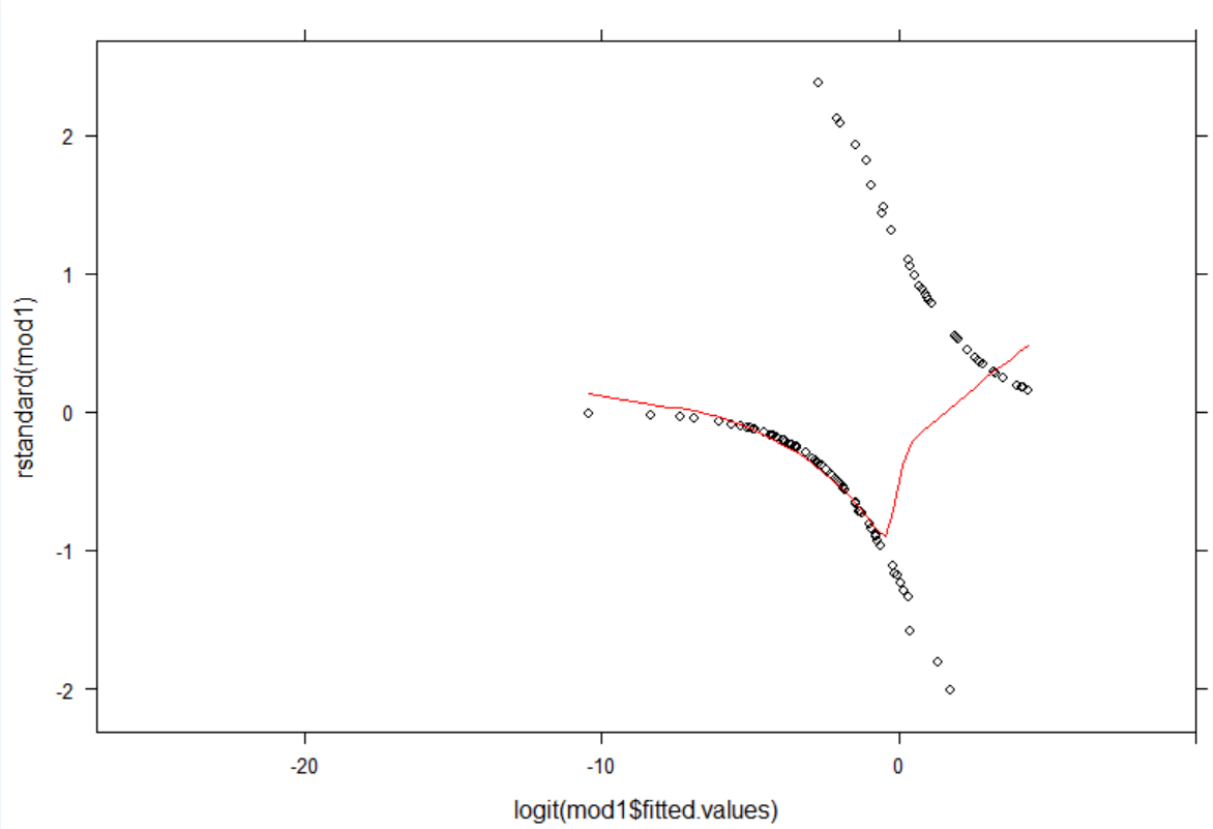
- Even though we don't write the error term down in the logit formulation of the regression model, the observed error (i.e., residual) is still there as always:

$$e_i = \text{observed success or failure} - \text{fitted probability of success}$$
$$= Y_i - \widehat{\Pr}(Y_i = 1)$$



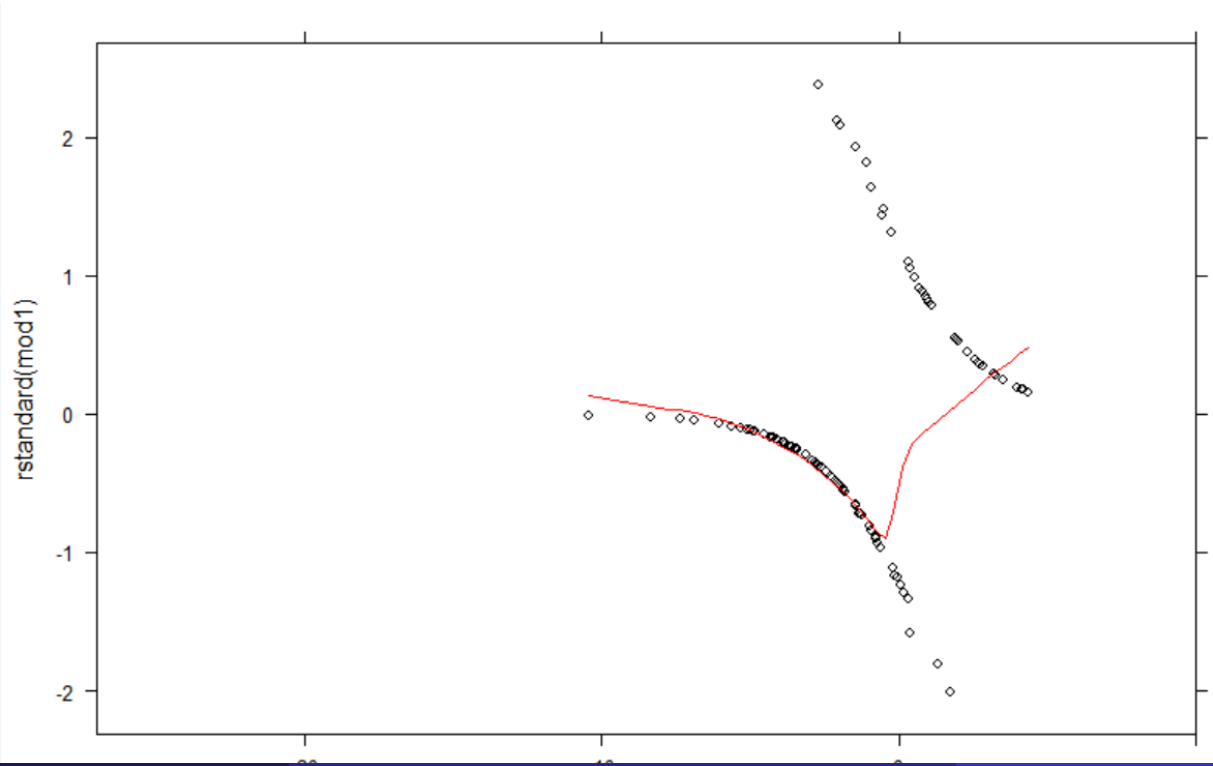
Logistic regression: An example

- Notice the binary structure of the residuals.
- A valid model will have errors that average to about zero everywhere.
- But no "equal variances" assumption; recall that for binary response data, *the variance is a function of the mean*.



Logistic regression: An example

- Still assuming independence of observations (can't usually be validated by a residual plot).
- *Very hard / impossible* to diagnose missing curvature, etc. from any residual diagnostic for binary data.
- My advice: Want the average residual always close to zero for as many fitted values as possible.



Logistic regression: An example

- Recall that I actually simulated these data, so I know the true data-generating process.
- Let's see what happens when we fit the *true* (i.e., correctly specified) model to the data:

```
Call:
glm(formula = y ~ x1 * x2 + x3 + I(x3^(1/3)), family = binomial(link = "logit"))

Deviance Residuals:
     Min        1Q    Median        3Q       Max
-1.80071  -0.32848  -0.04906   0.35852   2.68318

Coefficients:
            Estimate Std. Error z value Pr(>|z|)
(Intercept) -28.9307    37.8910  -0.764 0.445151
x1           -0.7739     0.7575  -1.022 0.306985
x2          -10.7435     2.8057  -3.829 0.000129 ***
x3           -3.2657     4.4789  -0.729 0.465916
I(x3^(1/3))  28.0202    35.2083   0.796 0.426123
x1:x2        -7.5109     2.8224  -2.661 0.007787 **
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

(Dispersion parameter for binomial family taken to be 1)

    Null deviance: 128.207  on 99  degrees of freedom
Residual deviance:  55.928  on 94  degrees of freedom
AIC: 67.928

Number of Fisher Scoring iterations: 7
```

Logistic regression: An example

- Reduction in residual deviance (and AIC).
- Estimated coefficients have changed, and in fact only two of them are significantly different from zero (low power).

```
Call:
glm(formula = y ~ x1 * x2 + x3 + I(x3^(1/3)), family = binomial(link = "logit"))

Deviance Residuals:
     Min        1Q    Median        3Q       Max
-1.80071  -0.32848  -0.04906   0.35852   2.68318

Coefficients:
             Estimate Std. Error z value Pr(>|z|)
(Intercept) -28.9307    37.8910  -0.764 0.445151
x1           -0.7739     0.7575  -1.022 0.306985
x2          -10.7435     2.8057  -3.829 0.000129 ***
x3           -3.2657     4.4789  -0.729 0.465916
I(x3^(1/3))  28.0202    35.2083   0.796 0.426123
x1:x2        -7.5109     2.8224  -2.661 0.007787 **
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

(Dispersion parameter for binomial family taken to be 1)

    Null deviance: 128.207  on 99  degrees of freedom
Residual deviance:  55.928  on 94  degrees of freedom
AIC: 67.928

Number of Fisher Scoring iterations: 7
```

Logistic regression: An example

- True data-generating process here:

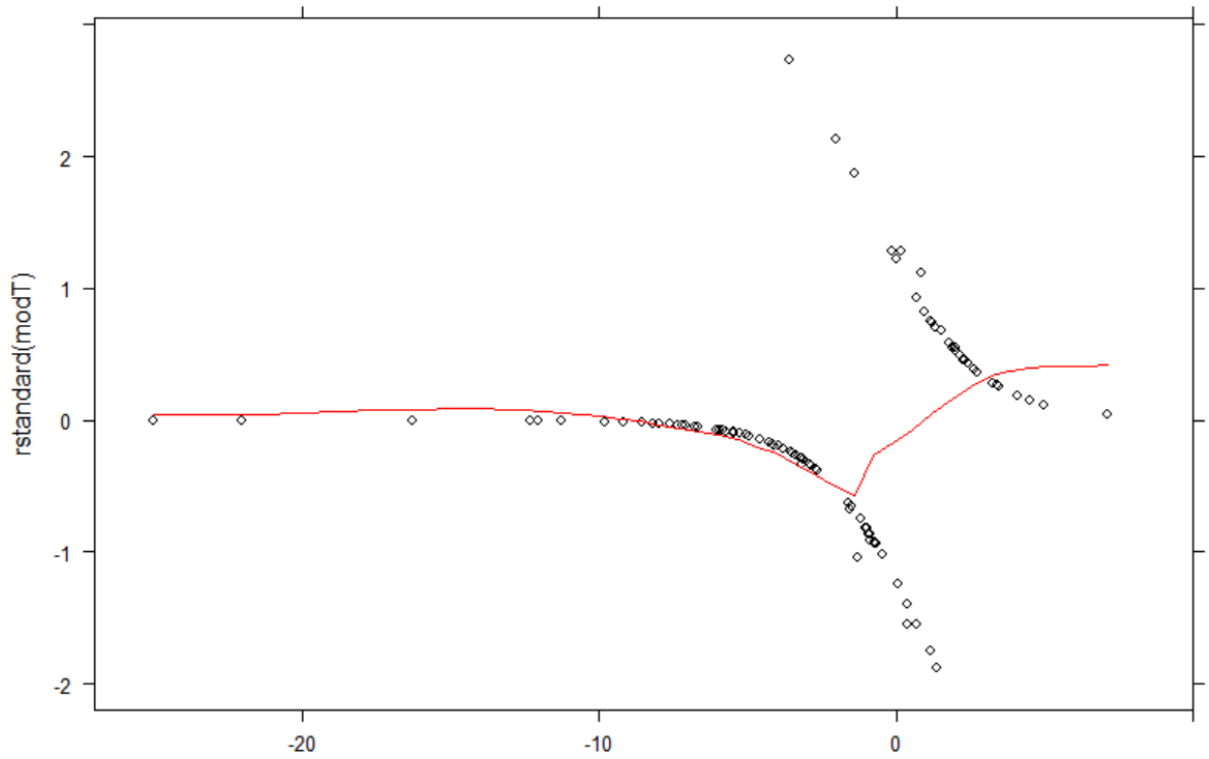
$$\Pr(Y = 1) = \frac{1}{1 + \exp(-(3 - 2X_1 - 9X_2 + X_3 - 5X_1X_2 - 3X_3^{1/3}))}$$

- Notice the poor quality of our estimates, even under correct model specification they hardly match their population values!
- This is due to the *low power* inherent to all forms of logistic regression.

```
Call:
glm(formula = y ~ x1 * x2 + x3 + I(x3^(1/3)), family = binomial(link = "logit"))

Deviance Residuals:
     Min        1Q    Median        3Q       Max
-1.80071  -0.32848  -0.04906   0.35852   2.68318

Coefficients:
             Estimate Std. Error z value Pr(>|z|)
(Intercept) -28.9307    37.8910  -0.764 0.445151
x1           -0.7739     0.7575  -1.022 0.306985
x2          -10.7435     2.8057  -3.829 0.000129 ***
x3           -3.2657     4.4789  -0.729 0.465916
I(x3^(1/3))  28.0202    35.2083   0.796 0.426123
x1:x2        -7.5109     2.8224  -2.661 0.007787 **
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

Logistic regression: An example

- Residual diagnostics look much better for this model though: average residual close to zero for more fitted values, and mean-deviation from zero is less extreme than before.
- Note though: Even for the correctly specified model the residual vs. fitted plot does not look perfect!



Logistic regression: Other issues

- Hopefully, it is now clear that logistic regression can be difficult to do well (just like ordinary linear regression)!
- Other issues we don't have time to explore:
  - Over/underdispersion of binary responses.
  - Quasi/full separation of binary responses over predictors.
- All these problems are even harder in multinomial regression contexts.
- Learn more about all these things (and more!) in EPSE 682.

Final thoughts: Linear regression analysis

- **There is no one, correct way to go about building a good regression model.** Lots of trial and error. Lots of exploration and iteration. Some good general things to do (or not do):

  (1) **Explore** the data and the observed relationships between variables.

  (2) Find a candidate set of **valid** models.

  (3) Among your candidates, find the **most efficient**/informative/best-fitting model.

- Remember: regression modelling captures *correlations* between variables only. Causal inference/modelling requires extra machinery (assumptions/structure/design).

Final thoughts: Linear regression analysis

(1) **Explore** the data and the observed relationships between variables.

- Plot the response vs. each predictor individually.
- For categorical predictors, examine the distribution of the response (i.e., histogram) over each factor level.
- Plot predictors against each other (e.g., scatterplots, histograms, boxplots).
- Perform small regressions with only one or two predictors to get a feel for how things correlate in a simplified domain.

Final thoughts: Linear regression analysis

(2) Find a candidate set of **valid** models.

- If any predictor variable(s) should be in your model because of prior knowledge and/or theoretical considerations, then *put them in the model* somewhere, some way.
- Do *not* remove predictors just because their estimated coefficients are not significant (i.e., not statistically distinguishable form zero) in one particular fitted model. Only possibly remove a predictor if its estimated effect *stabilizes* around zero (see below).
- As you consider more models (often of increasing complexity), you should notice your estimated effects *stabilize*, i.e., not change too much in value from one model to another. If this doesn't happen, then your model is likely severely misspecified and/or uninformative.

Final thoughts: Linear regression analysis

(2) Find a candidate set of **valid** models (i.e., obey all model assumptions).

- Always check residual vs. fitted values plots for clues about model misspecification.
- Remember: A good looking residual plot does *not* mean that you necessarily have the "correct" model. It is *diagnostic* only.
- Remember: With real data, you may not be able to find even a single totally valid model.
- Examine simpler models (e.g., marginal curvature in one variable at a time, or two predictor with an interaction model) to see if they explain some variation before building more complex models that incorporate these simpler models.
- For real research settings, be prepared to spend *many hours/days/weeks* considering different regression models.
- **NEVER** trust *automated model building* procedures unless you have huge datasets on relatively few predictor variables.

Final thoughts: Linear regression analysis

(3) Among your candidates, find the **most efficient**/informative/best-fitting model.

- Highly correlated predictors often just add a lot of noise (i.e., inflate standard errors) for minimal added information value.
- Always examine *residual standard error*; hopefully, it should *decrease* as your model improves.
- Use AIC/BIC to distinguish information content of similarly valid models and to avoid overfitting.
- Do not forget about parsimony and *ease of interpretation*; these are important (non-statistical) components of choosing the "best" model.
- Remember: Your estimated model coefficients will *change as you change the functional form of your model*. Hence, your estimated effects (and inferences) are always **model-dependent**, *not just data-dependent*.

Final thoughts: Linear regression analysis

- Thanks for your attention and good luck in your future work!
- Consider taking more stats courses, regardless of your research domain.
- Stats questions in the future, feel free to contact me (I can at least tell you if something is an easy issue/question or a hard one).