Multinomial Logistic Regression

You'll be formally learning about multinomial logistic regression today.

Previously, you used binomial logistic regression to do *two-class classification*, where you modeled the probability of a binary outcome as being linearly related to a number of predictor variables. The technique of *multinomial logistic regression* is a straightforward extension of this: our outcome variable has more than two categories, and we model the probability of falling into each category as being linearly related to our predictor variables.

Multinomial logistic regression is sometimes called *softmax regression*.[1]

Using multinomial logistic regression

You can use multinomial logistic regression with `glmnet(x, y, family="multinomial")`, where `x` is a scaled matrix of predictors and `y` is a numeric vector representing a categorical variable. In the following, you can just set `lambda=0`, because we aren't using very many predictors relative to the number of rows (so overfitting isn't a big problem).

Speed dating dataset

Return to the aggregated OkCupid dataset from Week 2.

- Use `table()` on the career code column to find the four most common listed careers in the dataset.
- Restricting to those four careers, predict career in terms of self-rated activity participation. Interpret the coefficients of the resulting linear model. Visualize them with `corrplot()`.

[1] This comes from the usage of the *softmax function*, which is a continuous approximation of the indicator function.

- Use your model to make predictions on the entire dataset and look at the principal components of the resulting log-odds ratios. Interpret the results.

OkCupid dataset

Think of one or two aspects of the OkCupid data to explore using multinomial logistic regression. Do so, incorporating your findings into your writeup of your results.