MY474: Applied Machine Learning for Social Science

Lecture 3: Logistic Regression

Blake Miller

21 October 2019

Agenda

1. `for` Loops
2. Visualizing data
3. OLS regression review
4. Logistic regression

`for` Loops

for Loop Example

A very simple for loop:

```r
for (i in 1:10) {
  print(i)
}
```

```
## [1] 1
## [1] 2
## [1] 3
## [1] 4
## [1] 5
## [1] 6
## [1] 7
## [1] 8
## [1] 9
## [1] 10
```

for Loop Example

A for loop that creates a vector of the squared value of numbers from one to ten and stores it in a vector result

```
a <- 1:10

result <- numeric(length = length(a))

for (i in seq_along(a)) {
  result[i] <- a[i]^2
}
result
```

```
##  [1]   1   4   9  16  25  36  49  64  81 100
```

Equivalent Vectorized Code

```r
sapply(1:10, function(x) x^2)
```

```
##  [1]   1   4   9  16  25  36  49  64  81 100
```

Equivalent Code

```
c(1:10)^2
```

```
##  [1]   1   4   9  16  25  36  49  64  81 100
```

Visualizing Data

Visualize Law of Large Numbers via flipping a fair coin

Flip a coin 1000 times

```
n <- 1000
x <- sample(0:1,n, repl=T)
head(x)
```

```
## [1] 0 0 0 0 1 1
```

Visualize Law of Large Numbers via flipping a fair coin

Count the number of cumulative heads

```
s <- cumsum(x)
head(x)
```

```
## [1] 0 0 0 0 1 1
```

```
head(s)
```

```
## [1] 0 0 0 0 1 2
```

```
tail(s)
```

```
## [1] 495 496 496 496 496 496
```

Visualize Law of Large Numbers via flipping a fair coin

Calculate the rate of heads at each interation of the simulation

```
r <- s/(1:n)
head(r, n=3)
```

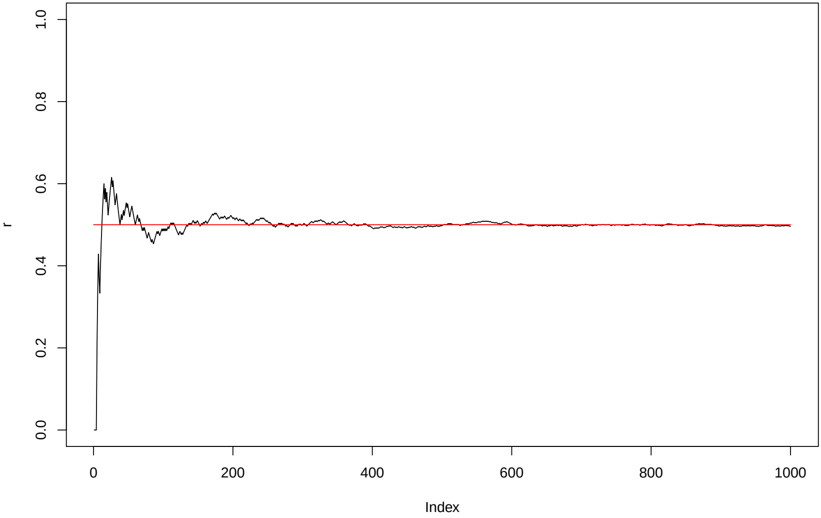
```
## [1] 0 0 0
```

```
tail(r, n=3)
```

```
## [1] 0.4969940 0.4964965 0.4960000
```

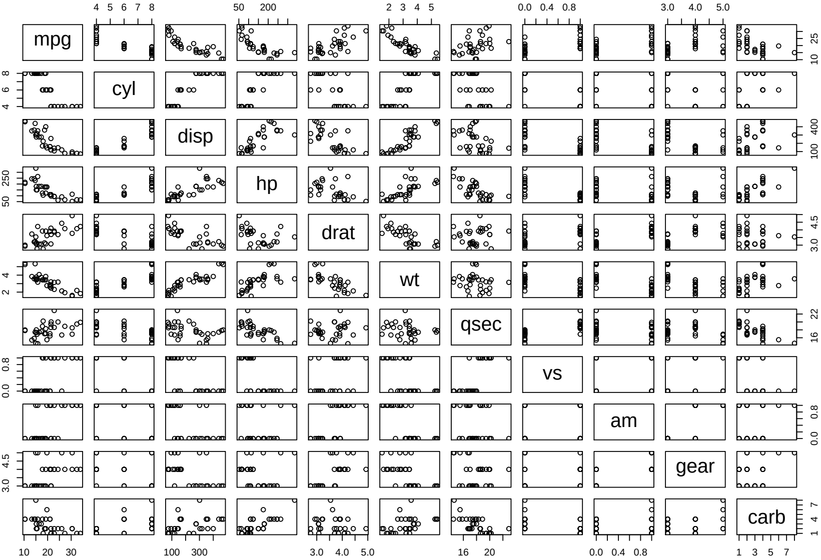
Visualize Law of Large Numbers via flipping a fair coin

Plot the rate at each iteration of the simulation and visualize the true rate in red.



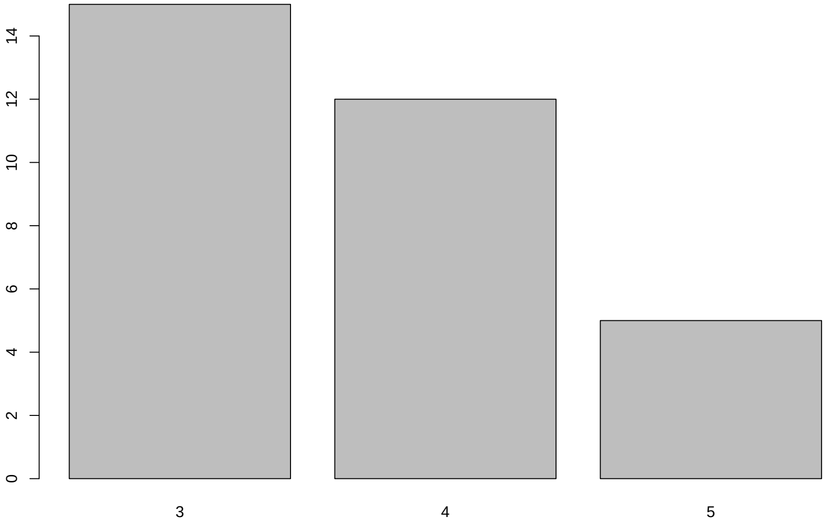
Visualizing Datasets: Pairs plots

```
pairs(mtcars)
```



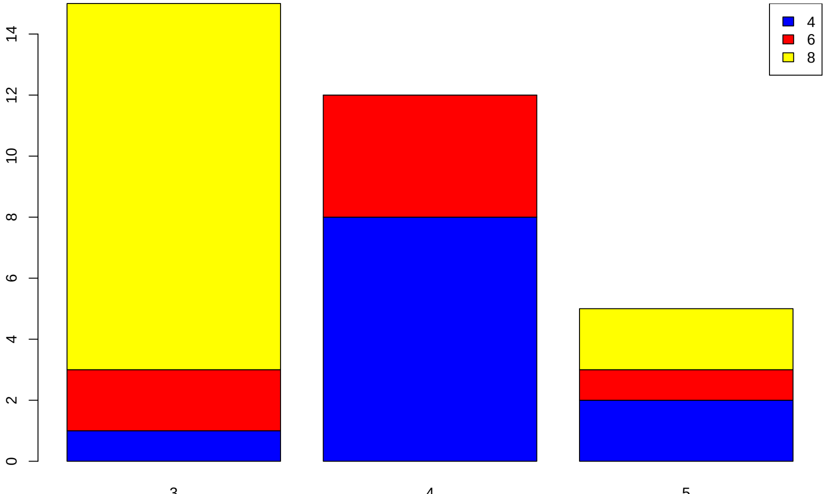
Visualizing Datasets: Barplots

```
gear_type <- table(mtcars$gear)
barplot(gear_type)
```



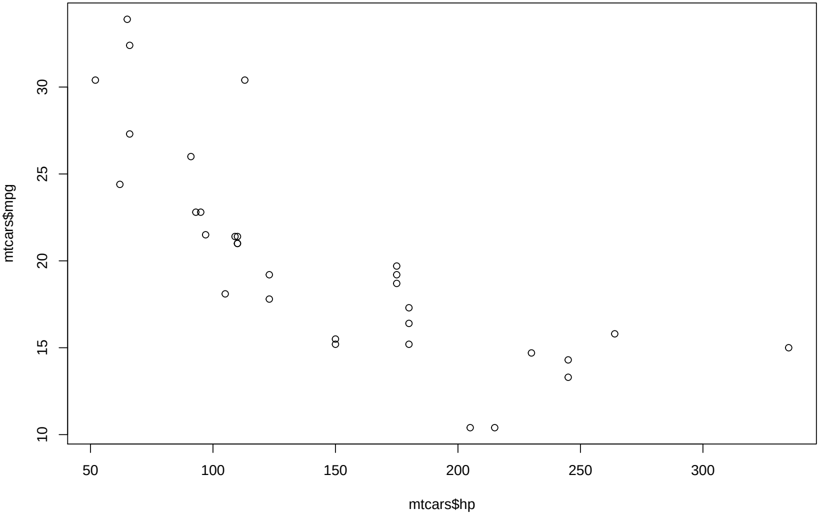
Visualizing Datasets: Stacked barplots

```r
gear_type_cyl <- table(mtcars$cyl, mtcars$gear)
colors = c("blue","red","yellow")
barplot(gear_type_cyl, col = colors)
legend("topright", rownames(gear_type_cyl), fill = colors)
```



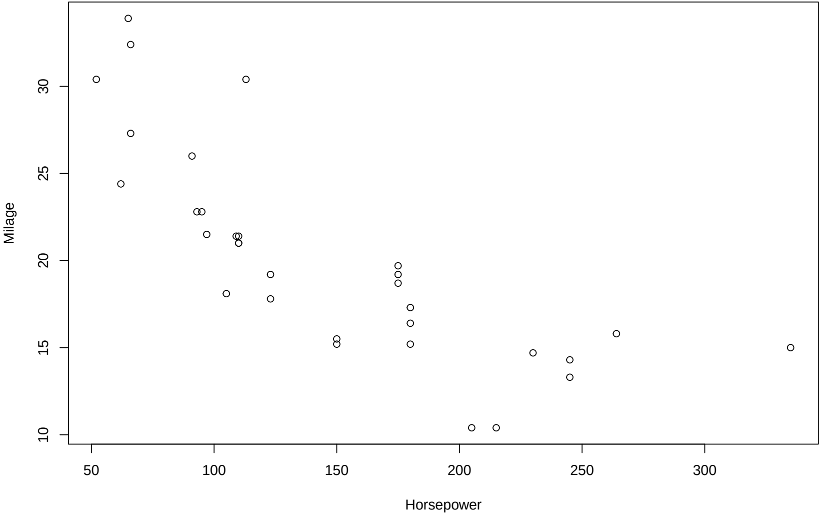
Visualizing Datasets: Scatterplots

```
plot(x = mtcars$hp, y = mtcars$mpg)
```



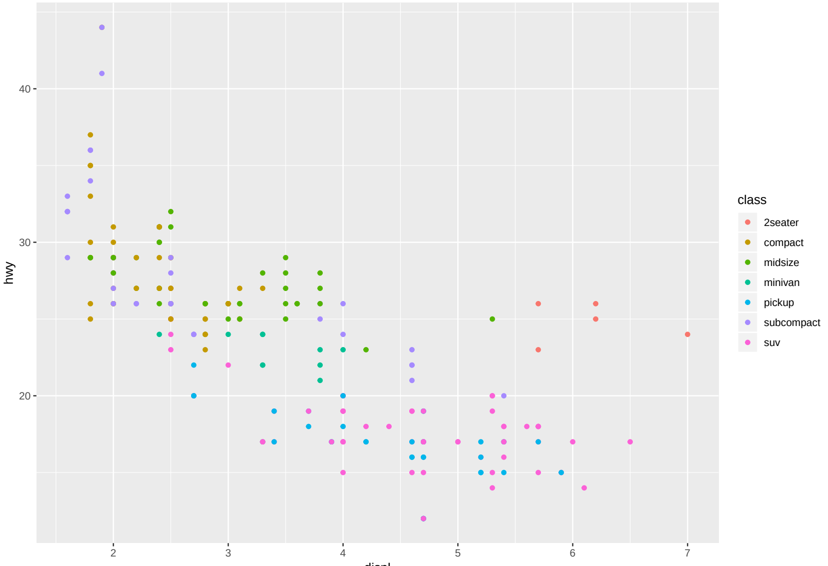
Visualizing Datasets: Axis Labels

```
plot(x = mtcars$hp, y = mtcars$mpg,
     xlab = "Horsepower",
     ylab = "Milage")
```



More customizable, prettier plots: ggplot2

```
ggplot(data=mpg, aes(x=displ, y=hwy, colour = class)) +
  geom_point()
```



OLS Regression

Simple Linear Regression Model

Assume the linear model

$$y_i = \beta_0 + \beta x_1 + \varepsilon_i, \quad \varepsilon_i \sim \mathcal{N}(0, \sigma^2), \quad i = 1, 2, \ldots, n$$

- y_i is the random response variable (observed data)
- x_i is the fixed predictor variable (observed data)
- β_0 is the fixed and unknown intercept parameter (not observed)
- β_1 is the fixed and unknown slope parameter (not observed)
- ε_i is the random error term (not observed)

Least Squares Criterion

To estimate the following model:

$$y_i = \beta_0 + \beta x_1 + \varepsilon_i$$

We minimize RSS, a **loss function** that is most commonly used to fit/train an OLS model:

$$\text{RSS} = \sum_{i=1}^{n} e_i^2 = \sum_{i=1}^{n} (y_i - \hat{y})^2 = \sum_{i=1}^{n} (y_i - \hat{\beta}_0 - \hat{\beta} x_i)^2$$

The **least squares criterion** is:

$$\underset{\hat{\beta}_0, \hat{\beta}_1}{\text{minimize}} \sum_{i=1}^{n} (y_i - \hat{\beta}_0 - \hat{\beta} x_i)^2$$

Estimation

Normal Equations

$$\frac{\partial \, \mathrm{RSS}}{\partial \beta_0} = -2(y_i - \beta_0 - \beta x_1) = 0$$
$$\frac{\partial \, \mathrm{RSS}}{\partial \beta_1} = -2(y_i - \beta_0 - \beta x_1) x_i = 0$$

Solutions

$$\hat{\beta}_0 = \bar{y} - \hat{\beta} \bar{x}$$
$$\hat{\beta}_1 = \frac{\sum_{i=1}^{n} (x_i - \bar{x}) y_i}{\sum_{i=1}^{n} (x_i - \bar{x})^2}$$

$$\bar{y} \equiv \tfrac{1}{n} \textstyle\sum_{i=1}^{n} y_i, \quad \bar{x} \equiv \tfrac{1}{n} \textstyle\sum_{i=1}^{n} x_i$$

The Normal Linear Model

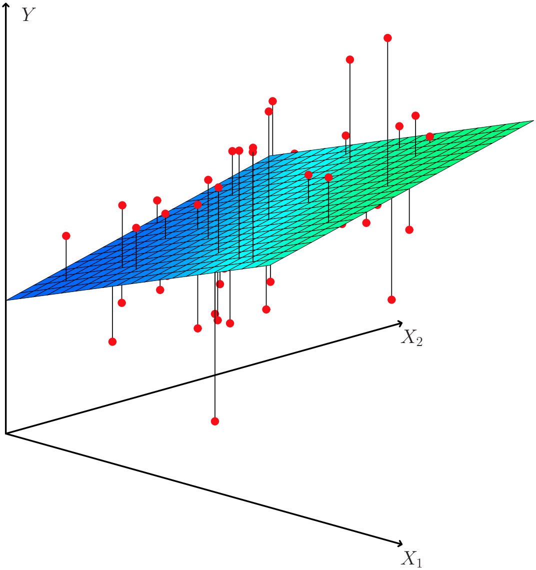
Assume the linear model

$$\mathbf{y} = \boldsymbol{X}\beta + \varepsilon, \quad \boldsymbol{\varepsilon} \sim \mathcal{N}_n(0, \sigma^2 \mathbf{I}_n)$$

- $\mathbf{y}$ is the $n \times 1$ random response vector (observed)
- $\mathbf{X}$ is the fixed $n \times p$ matrix of predictor variables (observed data)
- β is the $p \times 1$ vector of fixed and unknown parameters (not observed)
- ε is the $n \times 1$ vector of error terms (not observed)

$$\begin{bmatrix} y_1 \\ y_2 \\ \vdots \\ y_n \end{bmatrix} = \begin{bmatrix} X_{11} & X_{12} & \cdots & X_{1p} \\ X_{21} & X_{22} & \cdots & X_{2p} \\ \vdots & \vdots & \ddots & \vdots \\ X_{n1} & X_{n2} & \cdots & X_{np} \end{bmatrix} \begin{bmatrix} \beta_1 \\ \beta_2 \\ \vdots \\ \beta_p \end{bmatrix} + \begin{bmatrix} \varepsilon_1 \\ \varepsilon_2 \\ \vdots \\ \varepsilon_p \end{bmatrix}$$

Least Squares Estimation



Least Squares Estimation

- Find β to minimize

$$\text{RSS} = \varepsilon'\varepsilon = (\mathbf{y} - \mathbf{X}\beta)'(\mathbf{y} - \mathbf{X}\beta)$$

- Leads to the normal equations

$$(\mathbf{X}'\mathbf{X})\hat{\beta} = \mathbf{X}'\mathbf{y}$$

- Assuming **X** has rank p, the unique solution is

$$\hat{\beta} = (\mathbf{X}'\mathbf{X})^{-1}\mathbf{X}'\mathbf{y}$$

Fitted Values, Residuals, σ^2

- The fitted values are

$$\hat{\mathbf{y}} = \mathbf{X}\hat{\beta} = \mathbf{X}(\mathbf{X}'\mathbf{X})^{-1}\mathbf{X}'\mathbf{y}$$

- The residuals are

$$\begin{aligned}\mathbf{e} &= \mathbf{y} - \hat{\mathbf{y}} = \mathbf{y} - \mathbf{X}\hat{\beta} \\ &= \mathbf{y} - \mathbf{X}(\mathbf{X}'\mathbf{X})^{-1}\mathbf{X}'\mathbf{y}\end{aligned}$$

- An unbiased estimate of σ^2 is

$$\sigma^2 = \frac{\mathbf{e}'\mathbf{e}}{n-p} = \frac{(\mathbf{y} - \mathbf{X}\hat{\beta})'(\mathbf{y} - \mathbf{X}\hat{\beta})}{n-p} = \frac{RSS}{n-p}$$

Properties of Least Squares Estimates

- $E[\hat{\beta}] = \beta$
- $Var(\hat{\beta}) = \sigma^2(X'X)^{-1}$
- The Gauss-Markov theorem states that $\hat{\beta}$ is the best linear unbiased estimate (BLUE) of β (i.e. it has minimum variance in the class of unbiased estimators)
- Under the normal model, the least squares estimates equal the maximum likelihood estimates

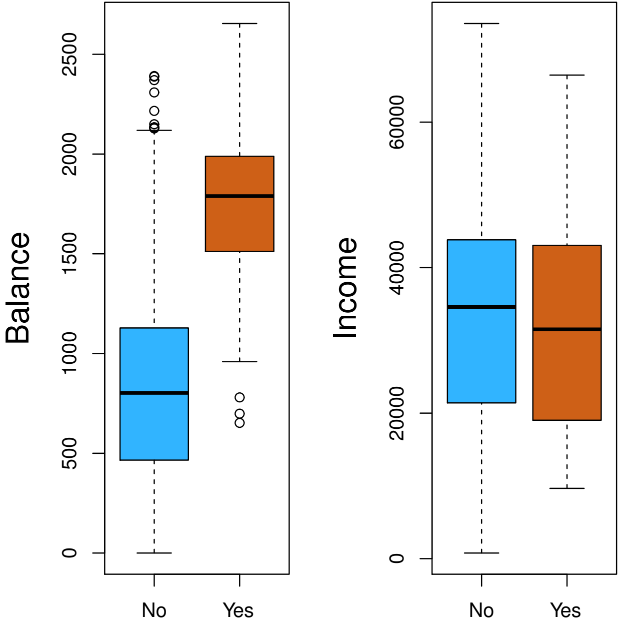
Interpreting regression coefficients

- The ideal scenario is when the predictors are uncorrelated - a **balanced design**
  - Each coefficient can be estimated and tested separately.
  - Interpretations such as "a unit change in X_j is associated with a β_j change in Y, while all the other variables stay fixed," are possible.
- Correlations amongst predictors cause problems:
  - The variance of all coefficients tends to increase, sometimes dramatically
  - Interpretations become hazardous — when X_j changes, everything else changes.
  - Recall the advertising data from the book. What if there is a fixed advertising budget? If more is spend on radio, less can be spent on TV, newspaper
- **Claims of causality** should be avoided for observational data.

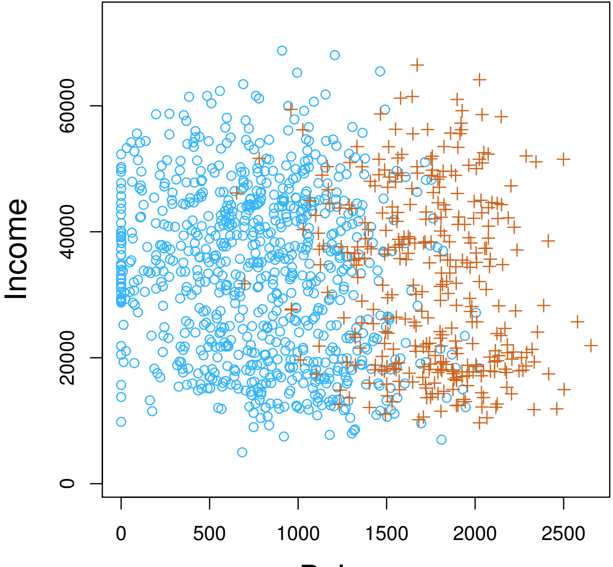
Classification

- Qualitative variables take values in an unordered set C, such as: eye color $\in \{brown, blue, green\}$ or email $\in \{spam, ham\}$.
- Given a feature vector X and a qualitative response Y taking values in the set C, the classification task is to build a function $C(X)$ that takes as input the feature vector X and predicts its value for Y ; i.e. $C(X) \in C$
- Often we are more interested in estimating the *probabilities* that X belongs to each category in C. For example, it is more valuable to have an estimate of the probability that an insurance claim is fraudulent, than a classification fraudulent or not.

Example: Credit Card Default



Example: Credit Card Default



Can we use Linear Regression?

Suppose for the Default classification task that we code

$$Y = \begin{cases} 0 & \text{if no} \\ 1 & \text{if yes} \end{cases}$$

- Can we simply perform a linear regression of Y on X and classify as Yes if $\hat{Y} > 0.5$?
  - In this case of a binary outcome, linear regression does a good job as a classifier, and is equivalent to **linear discriminant analysis** which we discuss later.
  - Since in the population $E(Y|X = x) = Pr(Y = 1|X = x)$, we might think that regression is perfect for this task.
  - However, **linear regression** might produce probabilities less than zero or bigger than one. **Logistic regression** is more appropriate.

Linear probability model

$$y = \beta_0 + \beta_1 X + \epsilon, \quad \varepsilon \overset{iid}{\sim} \mathcal{N}(0, \sigma^2)$$

$$y = \begin{cases} 0, & \text{not default} \\ 1, & \text{default} \end{cases}$$

- For a discrete random variable, we can express $E[Y|X]$ as a weighted sum:

$$\begin{aligned} E[Y|X] &= \sum_i P(y = y_i) y_i \\ &= p(y = 0|X) \times 0 + p(y = 1|X) \times 1 \\ &= p(y = 1|X) \end{aligned}$$

- So, $E[Y|X] = \beta_0 + \beta_1 X \quad \{(\textit{expectation of } E(\varepsilon) = 0)\}$

Problems with the linear probability model

- For the simple linear probability model, β_1 represents the change in $P(X)$ from a one unit increase in X, β_0 represents $P(y = 1|x = 0)$
- Consider the model with response `default` and predictor `balance`.
- $-\infty < \beta_0 + \beta_1 \times \text{balance} < \infty$
- This is a problem because $P(\text{default} = 1|\text{balance}) \in [0, 1]$
- The model is also heteroskedastic, non-normal errors

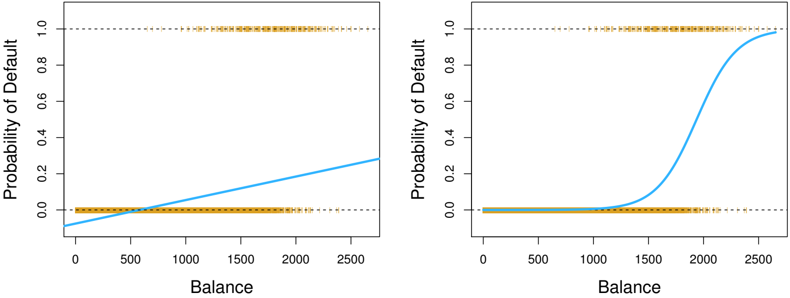
Nonlinear transformations of the linear probability model

- We can limit our dependent variable through a non-linear transform $F(\cdot)$ such that $P(\text{default} = 1|\text{balance}) = F(\beta_0 + \beta_1 \times \text{balance})$
- We should choose a function where $F(-\infty)$ approaches 0 and $F(\infty)$ approaches 1
- We could use the logistic function (among many other functions)

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

- $e \approx 2.71828$ is a mathematical constant called Euler's number.

Linear versus Logistic Regression



The orange marks indicate the response Y, either 0 or 1. Linear regression does not estimate $Pr(Y = 1|X)$ well. Logistic regression seems better suited to the task.

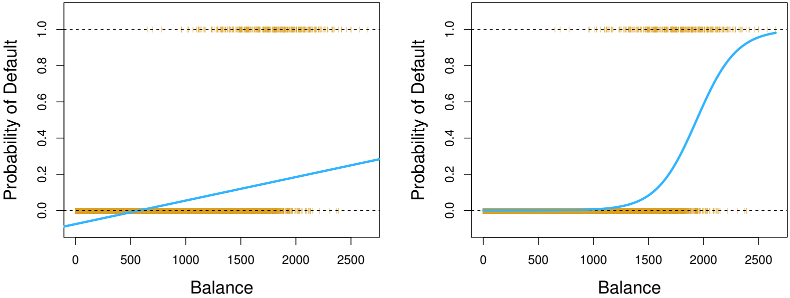
Logistic Regression

Let's write $p(X) = Pr(Y = 1|X)$ for short and consider using `balance` to predict `default`. Logistic regression uses the form

$$p(X) = \frac{e^{\beta_0+\beta_1 X}}{1 + e^{\beta_0+\beta_1 X}}$$

- No matter what values β_0, β_1 or X take, $p(X)$ will have values between 0 and 1. Why?
- This monotone transformation is called the log odds or logit transformation of $p(X)$.

Linear versus Logistic Regression



Logistic regression ensures that our estimate for p(X) lies between 0 and 1.

Odds ratios

- Ratio of success to failure
- Imagine a scenario with 0.8 probability of success, $p = 0.8$ and 0.2 probability of failure, $1 - p = 0.2$
- The odds ratio would be $\frac{0.8}{0.2} = 4$ which means a 4 to 1 odds of success

Odds ratios in logistic regression

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

$$\begin{aligned}
1-p &= 1-\frac{e^{\beta_0+\beta_1 X}}{1+e^{\beta_0+\beta_1 X}}\\
&= \frac{1+e^{\beta_0+\beta_1 X}}{1+e^{\beta_0+\beta_1 X}}-\frac{e^{\beta_0+\beta_1 X}}{1+e^{\beta_0+\beta_1 X}}\\
&= \frac{1}{1+e^{\beta_0+\beta_1 X}}
\end{aligned}$$

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

Log odds

- The linear combination of the predictors and the parameters β represent the log odds

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

- Interpreting logit coefficients is more tricky
- Increasing X by one unit changes the log odds by β_1 or multiplies the odds by e^{β_1}
- Because of non-linear relationship between X and $P(X)$, we cannot interpret β_1 as a a change in $P(X)$ associated with a one unit increase in X

Maximum Likelihood

We use maximum likelihood to estimate the parameters.

$$\ell(\beta_0, \beta_1) = \prod_{i:y_i=1} p(x_i) \prod_{i:y_i=0} (1 - p(x_i)).$$

- Joint probability of the observed zeros and ones in the data.
- Goal is to choose β_0 and β_1 to make the likelihood of the observed data as high as possible (because we **DID** observe these data!)
- Most statistical packages train linear logistic regression models by maximum likelihood. In `R` we use the `glm` function.

Making Predictions

```r
library(ISLR)
mod <- glm(default ~ balance, family = binomial,
           data = Default)
coef(mod)
```

```
##   (Intercept)       balance
## -10.651330614   0.005498917
```

Exercise 1: Making Predictions

$$\hat{p}(X) = \frac{e^{\hat{\beta}_0 + \hat{\beta}_1 X}}{1 + e^{\hat{\beta}_0 + \hat{\beta}_1 X}}$$

- What is the probability of default for someone with a balance of \$1000?
- What is the probability of default for someone with a balance of \$2000?
- What if we use student as the predictor? What is the probability of default for students and non-students?

Exercise 1: Answers

$$\hat{P}(\text{default} = 1|\text{balance} = 1000) = \frac{e^{-10.6513+0.0055\times1000}}{1 + e^{-10.6513+0.0055\times1000}} = 0.006$$

$$\hat{P}(\text{default} = 1|\text{balance} = 2000) = \frac{e^{-10.6513+0.0055\times2000}}{1 + e^{-10.6513+0.0055\times2000}} = 0.586$$

$$\hat{P}(\text{default} = 1|\text{student} = \text{Yes}) = \frac{e^{-3.5041+0.4049\times1}}{1 + e^{-3.5041+0.4049\times1}} = 0.0431$$

$$\hat{P}(\text{default} = 1|\text{student} = \text{No}) = \frac{e^{-3.5041+0.4049\times0}}{1 + e^{-3.5041+0.4049\times0}} = 0.0292$$

Logistic regression with several variables

$$log\left(\frac{p(X)}{1-p(X)}\right) = \beta_0 + \beta_1 X_1 + \cdots + \beta_p X_p$$

$$p(X) = \frac{e^{\beta_0+\beta_1 X_1+\cdots+\beta_p X_p}}{1+e^{\beta_0+\beta_1 X_1+\cdots+\beta_p X_p}}$$

Exercise 2: Multiple Logistic regression

1. Fit a logistic regression model with dependent variable `default` and independent variables `student`, `balance`, and `income` using the `glm()` function.
2. How are the coefficients different from the simple linear regression models we fit earlier?
3. What is the probability that a student with an income of \$10,000 and a balance of \$2500 defaults?

Exercise 2: Answers

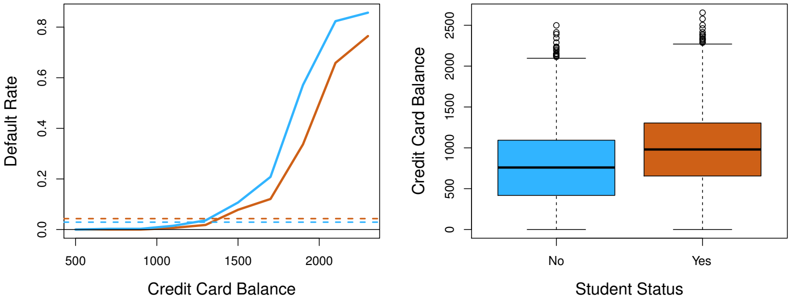
```r
mod <- glm(default ~ balance + student + income,
           family = binomial, data = Default)
round(coef(mod), 4)
```

```
## (Intercept)     balance  studentYes      income
##    -10.8690      0.0057     -0.6468      0.0000
```

```r
new_data <- list(balance=2500, income=10000,
                 student="Yes")
predict(mod, new_data, type="response")
```

```
##         1
## 0.9456165
```

Confounding



- Students tend to have higher balances than non-students, so their marginal default rate is higher than for non-students.
- But for each level of balance, students default less than non-students.
- Multiple logistic regression can tease this out.

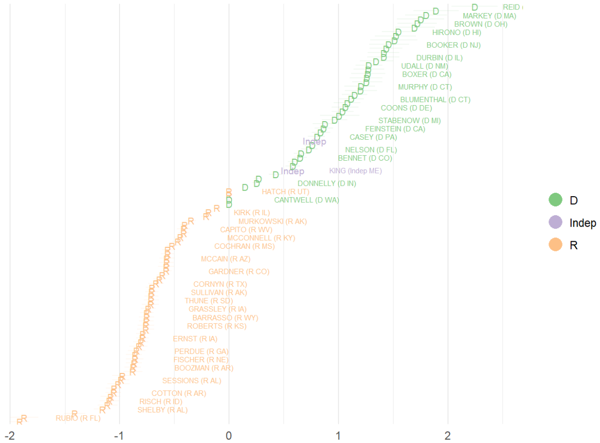
Logistic regression with more than two classes

So far we have discussed logistic regression with two classes. It is easily generalized to more than two classes. One version (used in the `R` package `glmnet`) has the symmetric form

$$P(Y = k|X) = \frac{e^{\beta_{0k}+\beta_{1k}X_1+\cdots+\beta_{pk}X_p}}{\sum_{\ell=1}^{K} e^{\beta_{0\ell}+\beta_{1\ell}X_1+\cdots+\beta_{p\ell}X_p}}$$

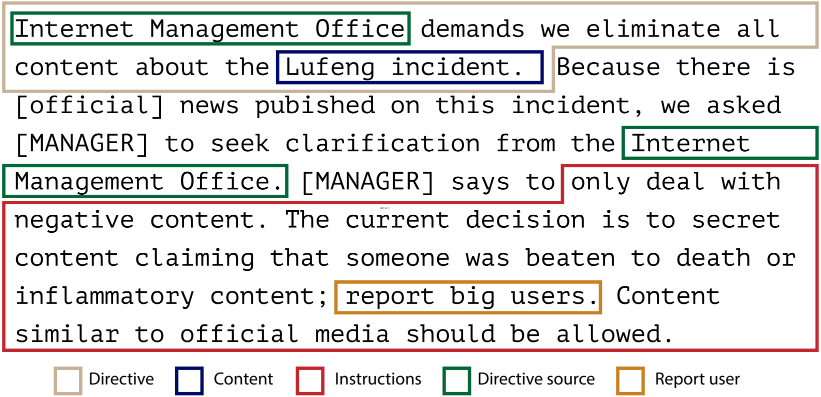
- Here there is a linear function for each class.
- Some cancellation is possible, and only K - 1 linear functions are needed as in 2-class logistic regression.
- Multiclass logistic regression is also referred to as multinomial regression.

Applications: Item Response Theory (IRT)



An extension of logistic regression can allow us to estimate ideal points fpr members of congress on certain bills. (Source: idealstan package)

Example: Leaked Censorship Logs

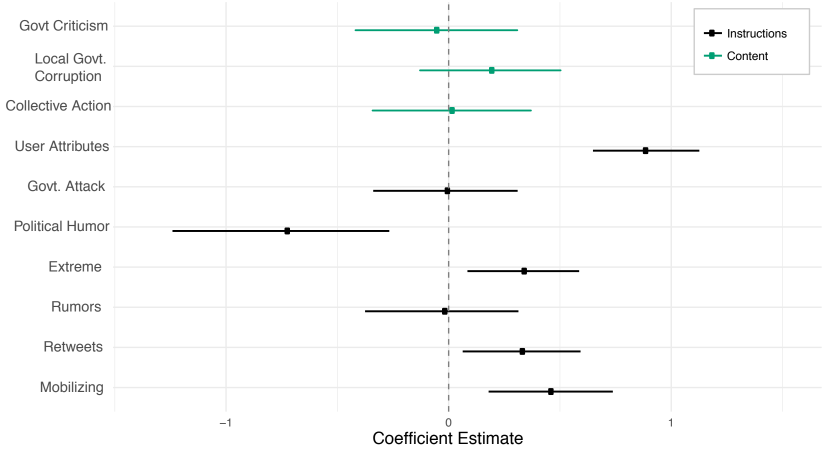


Internet Management Office demands we eliminate all content about the Lufeng incident. Because there is [official] news pubished on this incident, we asked [MANAGER] to seek clarification from the Internet Management Office. [MANAGER] says to only deal with negative content. The current decision is to secret content claiming that someone was beaten to death or inflammatory content; report big users. Content similar to official media should be allowed.

Example: Main Question

What kind of content and users are more likely to be reported back to the government?

Example: Leaked Censorship Logs


Govt Criticism
Local Govt. Corruption
Collective Action
User Attributes
Govt. Attack
Political Humor
Extreme
Rumors
Retweets
Mobilizing
Coefficient Estimate
Instructions
Content