

# STAT 213

## Logistic Regression I

Colin Reimer Dawson

Oberlin College

April 4-6, 2018

# Outline

Wrap Up: Model Selection  
Cross-Validation

Logistic Regression

# Outline

Wrap Up: Model Selection  
Cross-Validation

Logistic Regression

# Overfitting

- Too flexible model + fit procedure that maximizes fit to seen data = fitting “noise” as if it were “signal”
  - Extreme example: polynomial where number of coefficients equals number of observations
- This (mistaking noise for signal) is **overfitting**
- We need model selection techniques that protect against overfitting

## Scoring Criteria

1. Adj.  $R^2$ : balances fit and complexity
2. Mallows's  $C_p$  / Akaike Information Criterion (AIC): balances fit and complexity, using a “full” model as a reference
3. Out-of-sample predictive accuracy (estimate directly w/ cross-validation)

# Exploration Methods

1. Domain knowledge (+ a few  $F$ -tests)
2. Best subset
3. Forward selection
4. Backward selection
5. Stepwise selection

# Model Selection

Combines a scoring criterion with an exploration (“search”) method

		“Scoring”		
		$R_{adj.}^2$	$C_p$	CV Error
“Search”	Domain Knowledge			
	Best Subset			
	Forward Selection			
	Backward Selection			
	Stepwise Selection			

# Cross-Validation

**Validation** is a technique whereby the full dataset is divided into training and validation (held-out) sets. The first is used for fitting parameters; the second for evaluating predictive power.

**Cross-validation** uses all the data but gives each piece a turn as the validation set.

Versions:

1. Two-fold: Divide data (randomly) in half. Fit two models, exchanging roles of training and validation.
2.  $k$ -fold: Divide data into  $k$  equal sized sets, fit  $k$  models letting each set as the validation set.
3. Leave-one-out ( $n$ -fold): Let each observation be its own validation set. Requires fitting  $n$  models.



# Outline

Wrap Up: Model Selection  
Cross-Validation

Logistic Regression

# Quantitative Vs. Categorical Predictor and Response

		Response	
		Quantitative	Categorical
Predictor	Quantitative	Linear Reg.	Logistic Reg.
	Categorical	ANOVA	

# Logistic Regression

Handout

# Binary Logistic Regression

Response variable ( $Y$ ) is categorical with two categories (i.e., binary).

- Code  $Y$  as an indicator variable: 0 or 1
- Assume (for now) a single quantitative predictor,  $X$

# Two Equivalent Forms of Logistic Regression

Probability Form

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

Logit Form

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

$\pi$  : **Probability** that  $Y = 1$

$\frac{\pi}{1 - \pi}$  : **Odds** that  $Y = 1$

$\log\left(\frac{\pi}{1 - \pi}\right)$  : Log odds, or **logit** that  $Y = 1$

## Example: Golf Putts

Distance (ft)	3	4	5	6	7
# Made	84	88	61	61	44
# Missed	17	31	47	64	90
Total	101	119	108	125	134

1. Estimate the *probability* of success at each length
2. Estimate the *odds* of success at each length
3. Estimate the *log odds* of success at each length

# Golf Putts: Solutions

# Interpretations

- **Probability:** The long run proportion of “positive” to “total” instances
  - Scale: 0 to 1
  - As likely to happen as not =  $1/2$
- **Odds:** Ratio of “positive” to “negative” instances
  - Scale: 0 to  $\infty$
  - As likely as not = 1
- **Log Odds:** Log of the odds
  - Scale:  $-\infty$  to  $\infty$
  - As likely as not = 0
  - Symmetric, and unrestricted range



# Odds Ratios

## Logit and Odds

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

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

- In the model, for each 1 unit increase in  $X$ , the **log odds** increases by  $\beta_1$ .
- Equivalently: For each 1 unit increase in  $X$ , the **odds** are *multiplied* by  $e^{\beta_1}$
- In other words,  $e^{\beta_1}$  is the *odds ratio* resulting from a one unit change in  $X$ , with  $\beta_1$  the *log (of the) odds ratio*.

## Odds Ratios

The **odds ratio** associated with a binary response  $Y$  at two different predictor values  $X = x_2$  vs.  $X = x_1$  is the ratio of the odds. Denote by  $\pi(x)$  the probability of  $Y = 1$  when  $X = x$ . Then

$$\text{Odds Ratio}(x_2 \text{ vs. } x_1) = \frac{\pi(x_2)/(1 - \pi(x_2))}{\pi(x_1)/(1 - \pi(x_1))}$$

We can estimate this from a sample using:

$$\widehat{\text{Odds Ratio}}(x_2 \text{ vs. } x_1) = \frac{\hat{\pi}(x_2)/(1 - \hat{\pi}(x_2))}{\hat{\pi}(x_1)/(1 - \hat{\pi}(x_1))}$$

## Example: Golf Putts

Distance (ft)	3	4	5	6	7
# Made	84	88	61	61	44
# Missed	17	31	47	64	90
Total	101	119	108	125	134
$\hat{\pi}$	0.832	0.739	0.565	0.488	0.328
Odds	4.94	2.84	1.30	0.95	0.49
Log Odds	1.60	1.04	0.26	-0.05	-0.71

- Find the sample odds ratio for success for 4 ft. vs. 3 ft; 5 vs. 4; 6 vs. 5; 7 vs. 6
- Take the log of each of these to get the (additive) change in the logit. Should be slopes of lines “connecting the dots” (since  $\Delta X = 1$ ).

## Example: Golf Putts

Distance (ft)	3	4	5	6	7
# Made	84	88	61	61	44
# Missed	17	31	47	64	90
Odds	4.94	2.84	1.30	0.95	0.49
Log Odds	1.60	1.04	0.26	-0.05	-0.71
OR		0.575	0.457	0.734	0.513
$\Delta$ Log Odds		-0.56	-0.78	-0.31	-0.66

- In the data, successive ORs (changes in log odds) are different
- The model “smooths” the pattern

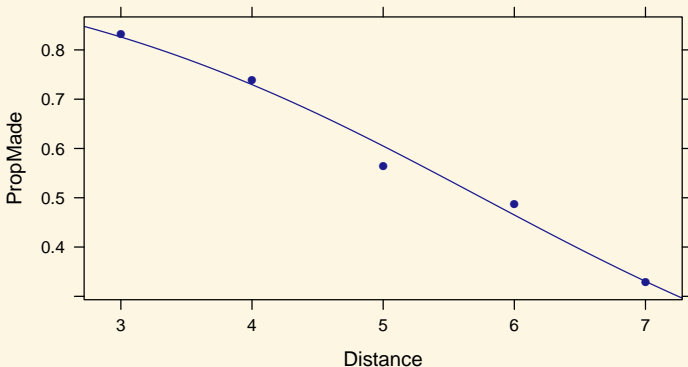
## Example: Golf Putts

```
library("mosaic")  
## This data is "binned" by x value. Usually won't be the case  
Putts <- data.frame(Distance = 3:7, Made = c(84,88,61,61,44),  
                    Total = c(101,119,108,125,134))  
Putts <- mutate(Putts, PropMade = Made / Total)  
## Version for binned data  
model <- glm(PropMade ~ Distance, weights = Total,  
             data = Putts, family = "binomial")  
model %>% coef() %>% round(2)
```

(Intercept)	Distance
3.26	-0.57

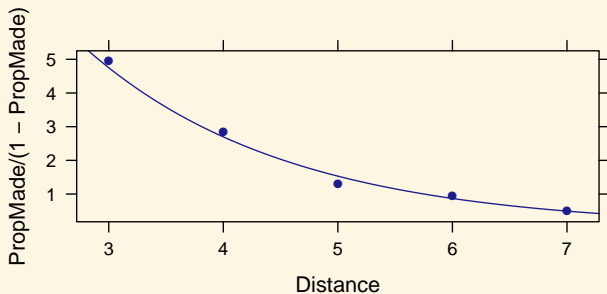
## Example: Golf Putts (Probabilities)

```
xyplot(PropMade ~ Distance, data = Putts)  
prob.hat <- makeFun(model)  
plotFun(prob.hat(Distance) ~ Distance, add = TRUE)
```



## Example: Golf Putts (Odds)

```
odds.hat <- makeFun(model, transformation = function(x){x/(1-x)})  
xyplot(PropMade/(1 - PropMade) ~ Distance, data = Putts)  
plotFun(odds.hat(Distance) ~ Distance, add = TRUE)
```

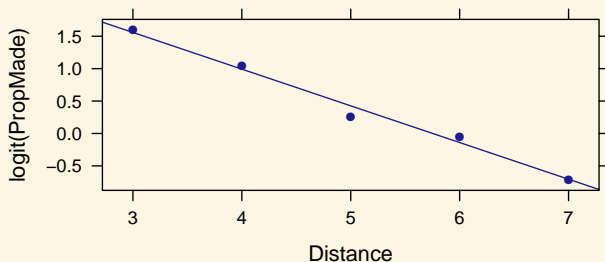


```
exp(-0.5661) ## Odds ratio for a one foot increase in Distance
```

```
[1] 0.5677353
```

## Example: Golf Putts (Log Odds)

```
log.odds.hat <- makeFun(model, transformation = logit)  
xyplot(logit(PropMade) ~ Distance, data = Putts)  
plotFun(log.odds.hat(Distance) ~ Distance, add = TRUE)
```



```
-0.5661 ## Log (odds ratio) / rate of change in log odds / slope of logit
```



## Reconstructing Odds Ratio

- The logistic regression output from R gives us  $\hat{\beta}_0$  and  $\hat{\beta}_1$ . But unlike in linear regression, these are not very interpretable on their own.
- We have seen that  $\beta_1$  corresponds to “rate of change in log odds”. (Slightly) better to convert to “odds ratio” per unit change in  $X$ .
- What do we do to  $\beta_1$  to get this?