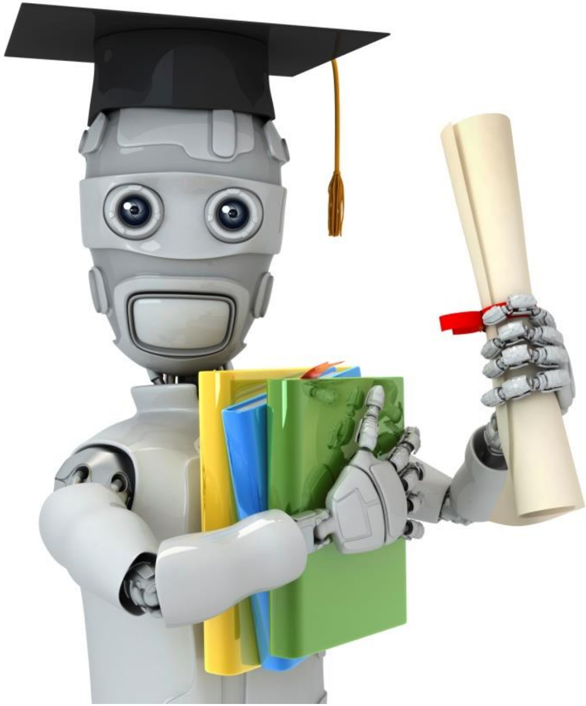
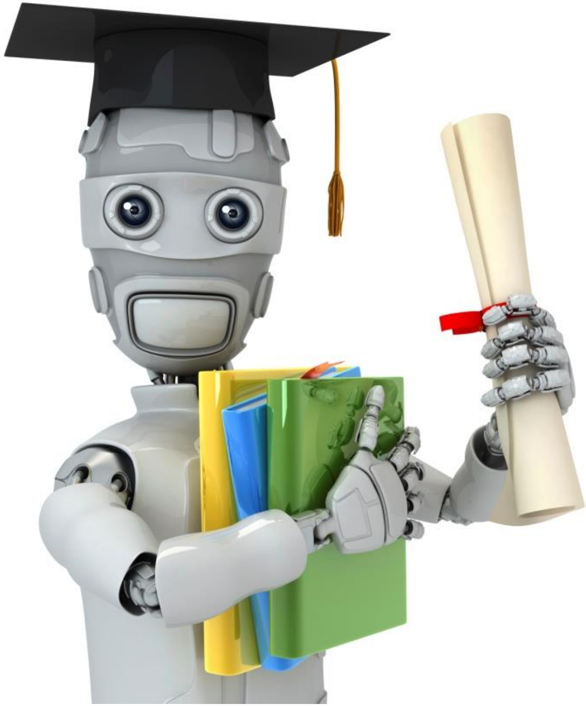


Machine Learning

Logistic regression

Fouad Hadj Selem



Machine Learning

Introduction and first example

Introduction

At this point we have covered:

- Simple linear regression
    - Relationship between numerical response and a numerical or categorical predictor
- Multiple regression
    - Relationship between numerical response and multiple numerical and/or categorical predictors

What we haven’t seen is what to do when the predictors are weird (nonlinear, complicated dependence structure, etc.) or when the response is weird (categorical, count data, etc.)

Example 1: Donner Party - Data

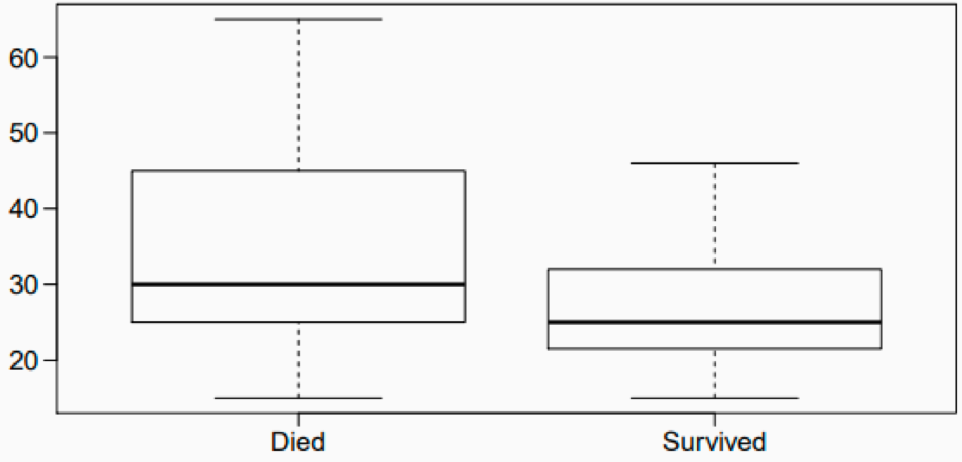
	Age	Sex	Status
1	23.00	Male	Died
2	40.00	Female	Survived
3	40.00	Male	Survived
4	30.00	Male	Died
5	28.00	Male	Died
⋮	⋮	⋮	⋮
43	23.00	Male	Survived
44	24.00	Male	Died
45	25.00	Female	Survived

Example 1: Donner Party Data

Status vs. Gender:

	Male	Female
Died	20	5
Survived	10	10

Status vs. Age:



Example 1: Donner Party Data

It seems clear that both age and gender have an effect on someone's survival, how do we come up with a model that will let us explore this relationship?

Even if we set Died to 0 and Survived to 1, this isn't something we can transform our way out of - we need something more.

One way to think about the problem - we can treat Survived and Died as successes and failures arising from a binomial distribution where the probability of a success is given by a transformation of a linear model of the predictors.

Generalized Linear Models

It turns out that this is a very general way of addressing this type of problem in regression, and the resulting models are called generalized linear models (GLMs). Logistic regression is just one example of this type of model.

All generalized linear models have the following three characteristics:

1. A probability distribution describing the outcome variable
2. A linear model
   - $\eta = \beta_0 + \beta_1 X_1 + \cdots + \beta_n X_n$
3. A link function that relates the linear model to the parameter of the outcome distribution
   - $g(p) = \eta$ or $p = g^{-1}(\eta)$

Logistic Regression

Logistic regression is a GLM used to model a binary categorical variable using numerical and categorical predictors.

We assume a binomial distribution produced the outcome variable and we therefore want to model p the probability of success for a given set of predictors.

To finish specifying the Logistic model we just need to establish a reasonable link function that connects η to p. There are a variety of options but the most commonly used is the logit function.

Logit function

$$logit(p) = \log\left(\frac{p}{1-p}\right), \text{ for } 0 \le p \le 1$$

Properties of the Logit

The logit function takes a value between 0 and 1 and maps it to a value between $-\infty$ and ∞.

Inverse logit (logistic) function

$$g^{-1}(x) = \frac{\exp(x)}{1 + \exp(x)} = \frac{1}{1 + \exp(-x)}$$

The inverse logit function takes a value between $-\infty$ and ∞ and maps it to a value between 0 and 1.

The Logistic Regression Model

The three GLM criteria give us:

$$y_i \sim \text{Binom}(p_i)$$

$$\eta = \beta_0 + \beta_1 x_1 + \cdots + \beta_n x_n$$

$$\text{logit}(p) = \eta$$

From which we arrive at,

$$p_i = \frac{\exp(\beta_0 + \beta_1 x_{1,i} + \cdots + \beta_n x_{n,i})}{1 + \exp(\beta_0 + \beta_1 x_{1,i} + \cdots + \beta_n x_{n,i})}$$

Training the Logistic Regression Model Using Cross Entropy Loss :1D Case

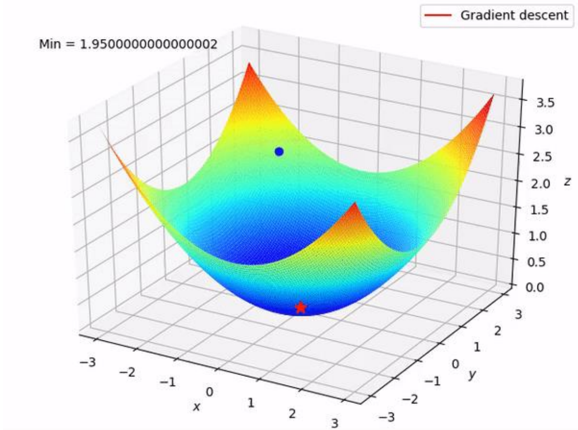
https://towardsdatascience.com/animations-of-logistic-regression-with-python-31f8c9cb420

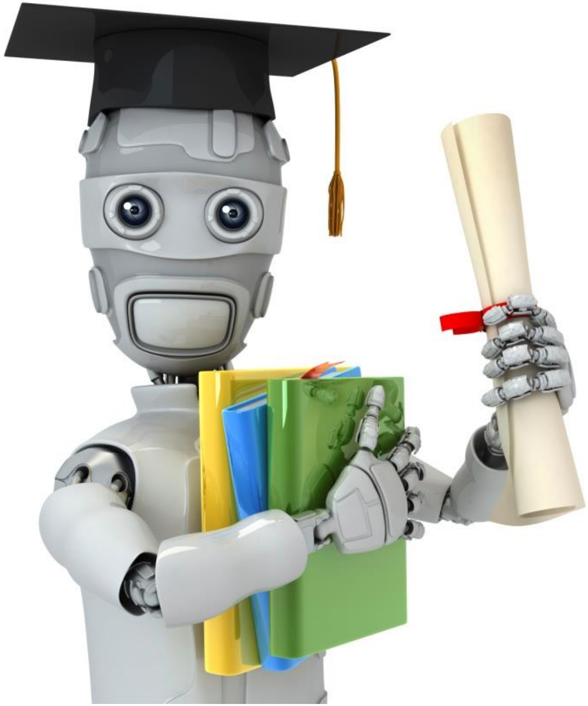
$$p(X) = \frac{e^{\beta_0+\beta_1 X}}{1+e^{\beta_0+\beta_1 X}}$$

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

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

$$l(\beta_0, \beta_1) = \prod_{i=1}^{n} p(x_i)^{Y_i}(1-p(x_i))^{1-Y_i}.$$





Logistic Regression in R

Machine Learning

Logistic Regression in R : Donner Data

In R we fit a GLM in the same was as a linear model except using `glm` instead of `lm` and we must also specify the type of GLM to fit using the `family` argument.

```
summary(glm(Status ~ Age, data=donner, family=binomial))

## Call:
## glm(formula = Status ~ Age, family = binomial, data = donner)
##
## Coefficients:
##              Estimate Std. Error z value Pr(>|z|)
## (Intercept)   1.81852    0.99937   1.820   0.0688 .
## Age          -0.06647    0.03222  -2.063   0.0391 *
##
##     Null deviance: 61.827  on 44  degrees of freedom
## Residual deviance: 56.291  on 43  degrees of freedom
## AIC: 60.291
```

Prediction

	Estimate	Std. Error	z value	Pr(>|z|)
(Intercept)	1.8185	0.9994	1.82	0.0688
Age	-0.0665	0.0322	-2.06	0.0391

Model:

$$\log\left(\frac{p}{1-p}\right) = 1.8185 - 0.0665 \times \text{Age}$$

Odds / Probability of survival for a newborn (Age=0):

$$\log\left(\frac{p}{1-p}\right) = 1.8185 - 0.0665 \times 0$$

$$\frac{p}{1-p} = \exp(1.8185) = 6.16$$

$$p = 6.16/7.16 = 0.86$$

Prediction 2

Model:

$$\log\left(\frac{p}{1-p}\right) = 1.8185 - 0.0665 \times \text{Age}$$

Odds / Probability of survival for a 25 year old:

$$\log\left(\frac{p}{1-p}\right) = 1.8185 - 0.0665 \times 25$$

$$\frac{p}{1-p} = \exp(0.156) = 1.17$$

$$p = 1.17/2.17 = 0.539$$

Odds / Probability of survival for a 50 year old:

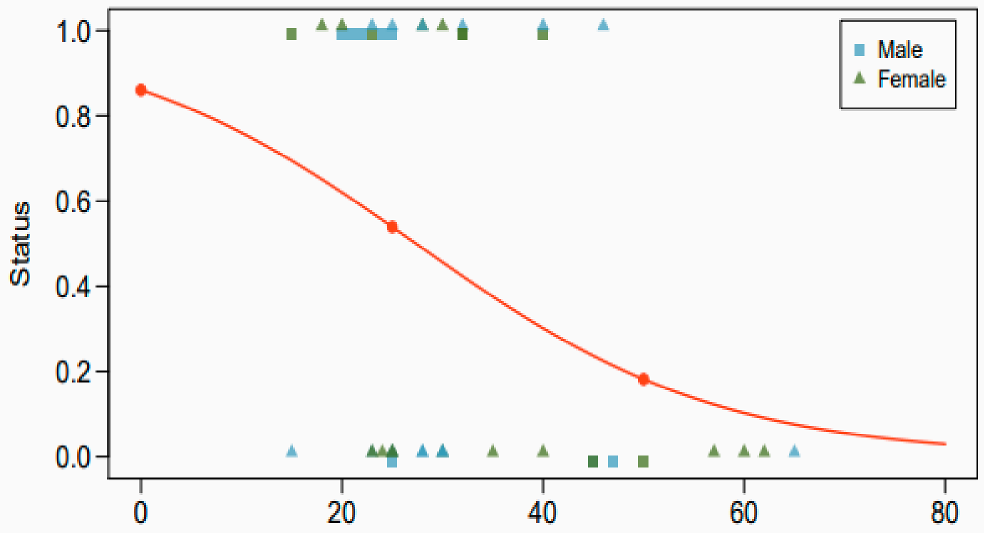
$$\log\left(\frac{p}{1-p}\right) = 1.8185 - 0.0665 \times 0$$

$$\frac{p}{1-p} = \exp(-1.5065) = 0.222$$

$$p = 0.222/1.222 = 0.181$$

Probabilities curve

$$\log\left(\frac{p}{1-p}\right) = 1.8185 - 0.0665 \times \text{Age}$$



With Two Variables (or more)

```
summary(glm(Status ~ Age + Sex, data=donner, family=binomial))

## Call:
## glm(formula = Status ~ Age + Sex, family = binomial, data = donner)
##
## Coefficients:
##             Estimate Std. Error z value Pr(>|z|)
## (Intercept)  1.63312    1.11018   1.471   0.1413
## Age         -0.07820    0.03728  -2.097   0.0359 *
## SexFemale    1.59729    0.75547   2.114   0.0345 *
## ---
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 61.827  on 44  degrees of freedom
## Residual deviance: 51.256  on 42  degrees of freedom
## AIC: 57.256
##
## Number of Fisher Scoring iterations: 4
```

Note: The model output does not include any F-statistic, as a general rule there are not single model hypothesis tests for GLM models.

We are however still able to perform inference on individual coefficients, the basic setup is exactly the same as what we've seen before except we use a Z test.

Probabilities Estimation

Just like MLR we can plug in gender to arrive at two status vs age models for men and women respectively.

General model:

$$\log\left(\frac{p_1}{1-p_1}\right) = 1.63312 + -0.07820 \times \text{Age} + 1.59729 \times \text{Sex}$$

Male model:

$$\begin{aligned}\log\left(\frac{p_1}{1-p_1}\right) &= 1.63312 + -0.07820 \times \text{Age} + 1.59729 \times 0 \\ &= 1.63312 + -0.07820 \times \text{Age}\end{aligned}$$

Female model:

$$\begin{aligned}\log\left(\frac{p_1}{1-p_1}\right) &= 1.63312 + -0.07820 \times \text{Age} + 1.59729 \times 1 \\ &= 3.23041 + -0.07820 \times \text{Age}\end{aligned}$$

Testing for the Slope of Age

	Estimate	Std. Error	z value	Pr(>|z|)
(Intercept)	1.6331	1.1102	1.47	0.1413
Age	-0.0782	0.0373	-2.10	0.0359
SexFemale	1.5973	0.7555	2.11	0.0345

$$H_0 : \beta_{age} = 0$$

$$H_A : \beta_{age} \neq 0$$

$$Z = \frac{\hat{\beta_{age}} - \beta_{age}}{SE_{age}} = \frac{-0.0782 - 0}{0.0373} = -2.10$$

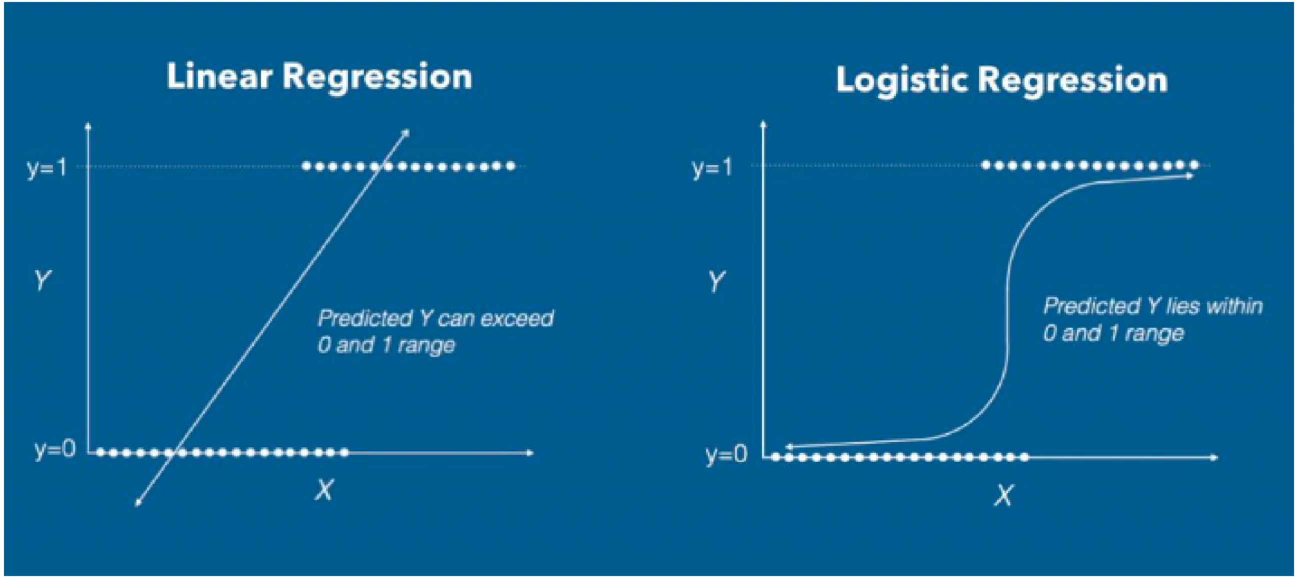
$$\text{p-value} = P(|Z| > 2.10) = P(Z > 2.10) + P(Z < -2.10)$$

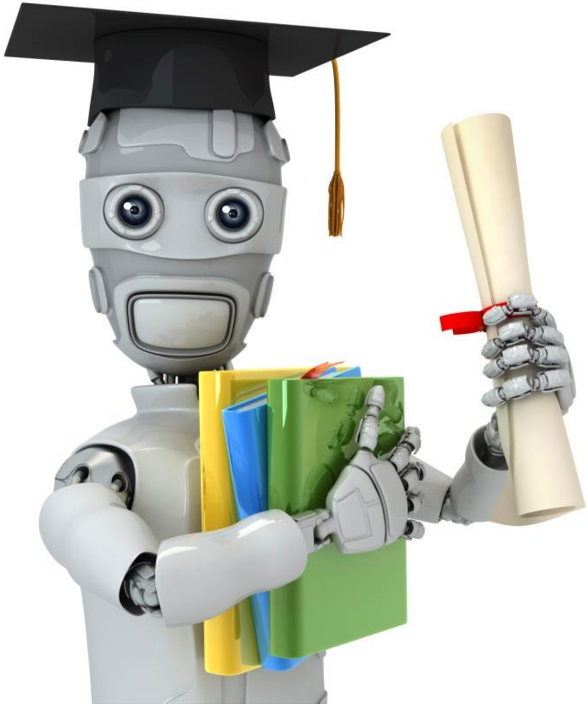
$$= 2 \times 0.0178 = 0.0359$$

Why not a Linear Regression?

When the response variable has only 2 possible values, it is desirable to have a model that predicts the value either as 0 or 1 or as a probability score that ranges between 0 and 1.

Linear regression does *not* have this capability. Because, If you use linear regression to model a binary response variable, the resulting model may not restrict the predicted Y values within 0 and 1.



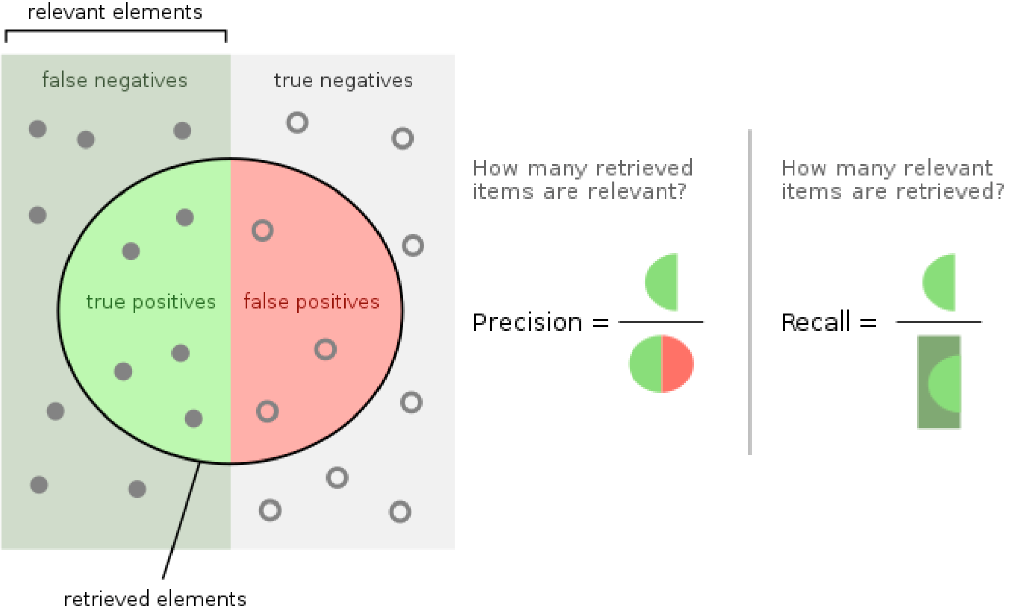


Machine Learning

Metrics Based on the Predicted Categories:

Classification Scores

		Predicted condition	
	Total population = P + N	Positive (PP)	Negative (PN)
Actual condition	Positive (P)	True positive (TP)	False negative (FN)
	Negative (N)	False positive (FP)	True negative (TN)



Mistakes have different costs:

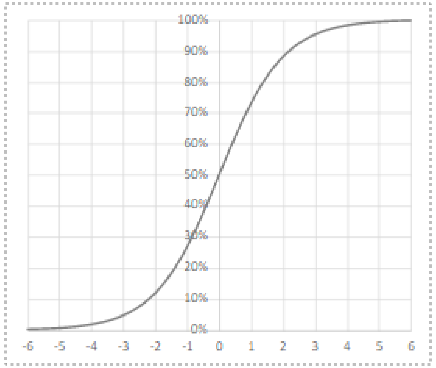
- **Disease Screening – LOW FN Rate**
- **Spam filtering – LOW FP Rate**

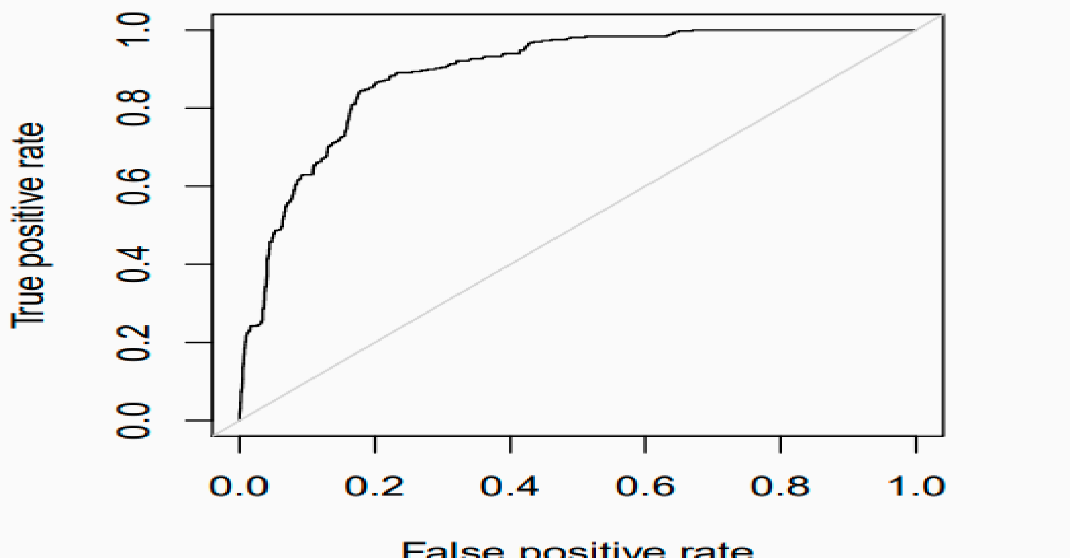
Conservative vs Aggressive settings:

- **The same application might need multiple tradeoffs**

Vary Threshold from 0.5?

- Logistic regression produces a score between 0 – 1 (probability estimate)
- Use threshold to produce classification
- What happens if you vary the threshold?

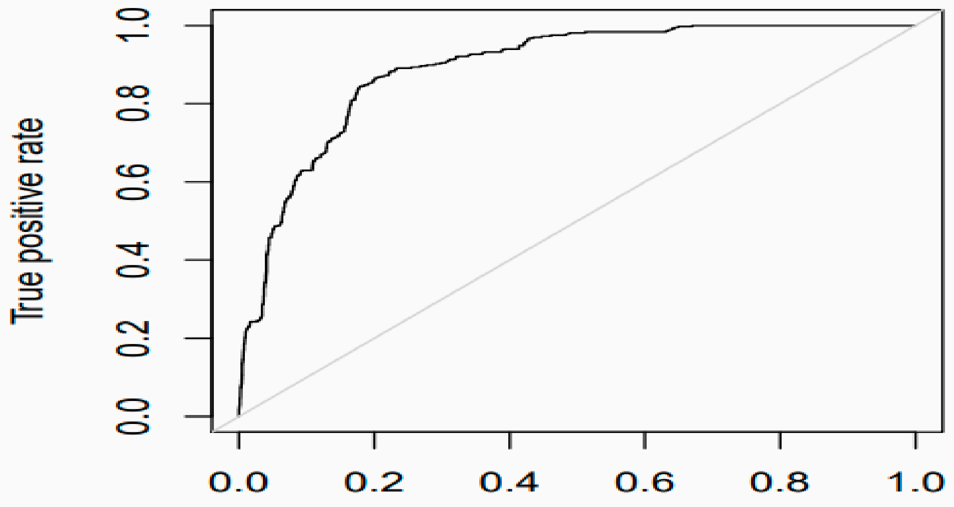




Receiver Operating Characteristic (ROC) curve.

Why do we care about ROC curves?

- Shows the trade off in sensitivity and specificity for all possible thresholds.
- Straight forward to compare performance vs. chance.
- Can use the area under the curve (AUC) as an assessment of the predictive ability of a model.



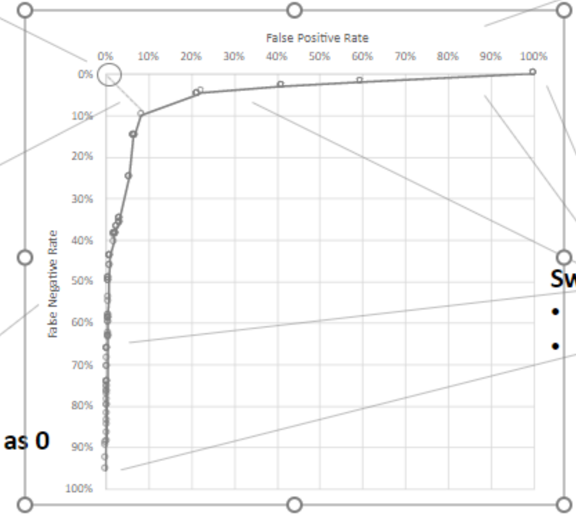
Checking Conditions for the Restaurant Tip Data

ROC Curve (Receiver Operating Characteristic)

Perfect score:
- **0% of 1s called 0**
- **0% of 0s called 1**

Percent of 0s classified as 1

This model's distance from perfect



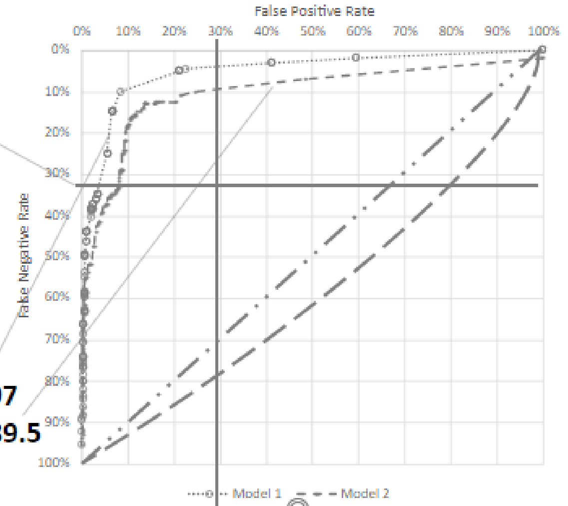
Sweep threshold from 0 to 1
- **Threshold 0: 'all' classified as 1**
- **Threshold 1: 'all' classified as 0**

Percent of 1s classified as 0

Comparing Models with ROC Curves

Model 1 better than Model 2 at every FPR or FNR target

Model 1 better than Model 2 at this FNR

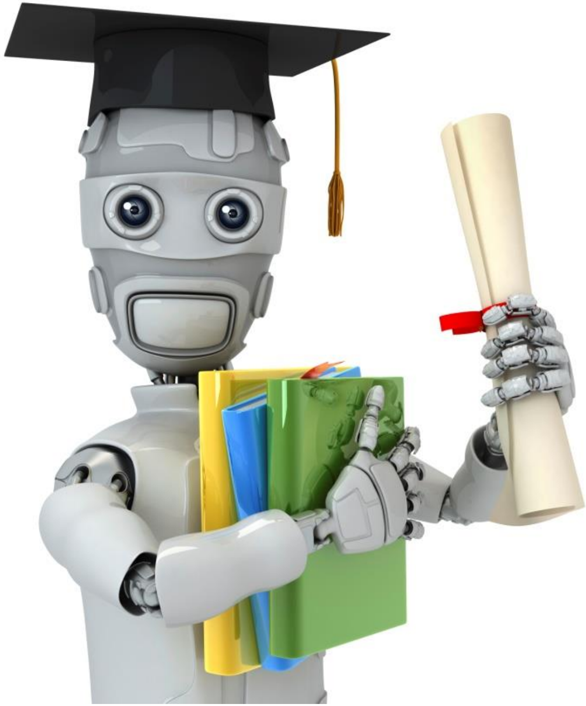


Model 1: AUC ~97
Model 2: AUC ~89.5

Model 1 better than Model 2

Area Under Curve (AUC)

- Integrate Area under the curve
- Perfect score is 1
- Higher scores allow for generally better tradeoffs
- AUC of 0.5 indicates model is essentially randomly guessing
- AUC of < 0.5 indicates you're doing something wrong…



Variable Selection for Logistic Regression

Machine Learning

Data and full model

We can begin with the full model. Full model can be denoted by using symbol "." on the right hand side of formula.

```
> library(MASS)
> head(bwt)
  low age lwt race  smoke ptd   ht    ui    ftv
1 0   19  182 black FALSE FALSE FALSE TRUE  0
2 0   33  155 other FALSE FALSE FALSE FALSE 2+
3 0   20  105 white TRUE  FALSE FALSE FALSE 1
4 0   21  108 white TRUE  FALSE FALSE TRUE  2+
5 0   18  107 white TRUE  FALSE FALSE TRUE  0
6 0   21  124 other FALSE FALSE FALSE FALSE 0

> full <- glm(low ~ ., family = binomial, data = bwt)
> summary(full)
Call:
glm(formula = low ~ ., family = binomial, data = bwt)
Deviance Residuals:
Min        1Q        Median    3Q       Max
-1.7038    -0.8068   -0.5008   0.8835   2.2152
```

Backward Selection

```
> backward<-stepAIC(full, direction="backward",trace =
FALSE)
> backward$anova
```

Stepwise Model Path

Analysis of Deviance Table

Initial Model:

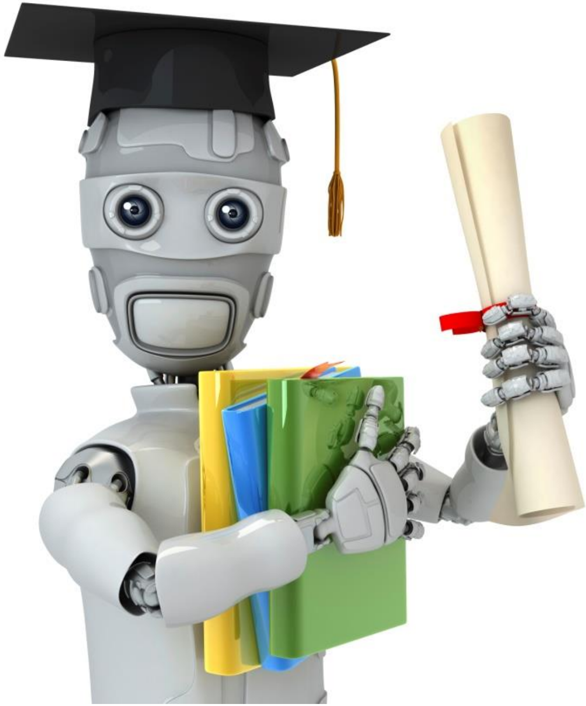
low ~ age + lwt + race + smoke + ptd + ht + ui + ftv

Final Model:

low ~ lwt + race + smoke + ptd + ht + ui

Step	Df	Deviance	Resid. Df	Resid. Dev	AIC
1			178	195.4755	217.4755
2 –ftv	2	1.358185	180	196.8337	214.8337
3 –age	1	1.017866	181	197.8516	213.8516

The backward elimination procedure eliminated variables *ftv* and *age*, which is exactly the same as the "both" procedure.



Machine Learning

See you next chapter

Thanks :)