Predicting Binary Response with Neural Networks

Professor Song Yao
Olin Business School

Customer Analytics

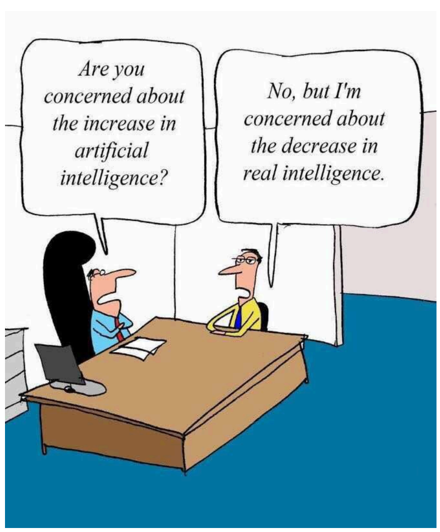
A brief introduction (very) to Neural Net

So far we have focused in heuristic and analyst-driven models

TYPES OF PREDICTIVE MODELS

- Heuristics (rules of thumb)
  - RFM Analysis
- Analyst-driven models (Statistical)
  - Regression models
  - Discrete choice models
- Data-driven models (Machine Learning)
  - Neural Networks (NN)
  - Decision Trees

Jokes about Machine Learning Models



Heuristic and analyst driven models each have negatives and positives

CHARACTERISTICS OF MODELING APPROACHES

RFM:	Logistic/Linear Regression:
- Simple and intuitive - Effective - Inexpensive - No special people or software needed **but...** - Does not "scale" well to include other variables - "Scores" customers only as member of a cell, not individually	- Easy to understand - Can incorporate just about any independent variable - Can test for significance and importance - "Scores" each customer individually **but...** - The model is only as good as your mental model of the process

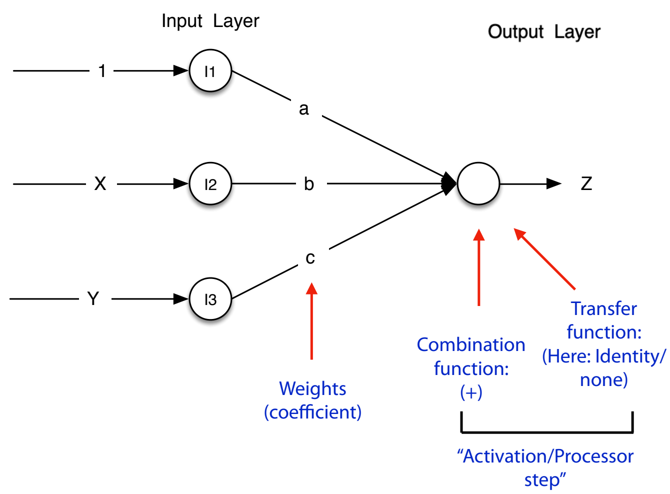
Fundamental Concept of Artificial Neural Network Models

- Artificial Neural Network (ANN) models are trying to simulate the brain network
  - Continuous and discrete output
- Human brain cells have
  - Have multiple inputs
  - Interact with one another
  - Have multiple outputs

Any regression equation can be expressed as a neural network

NEURAL NET VERSION OF A LINEAR REGRESSION EQUATION

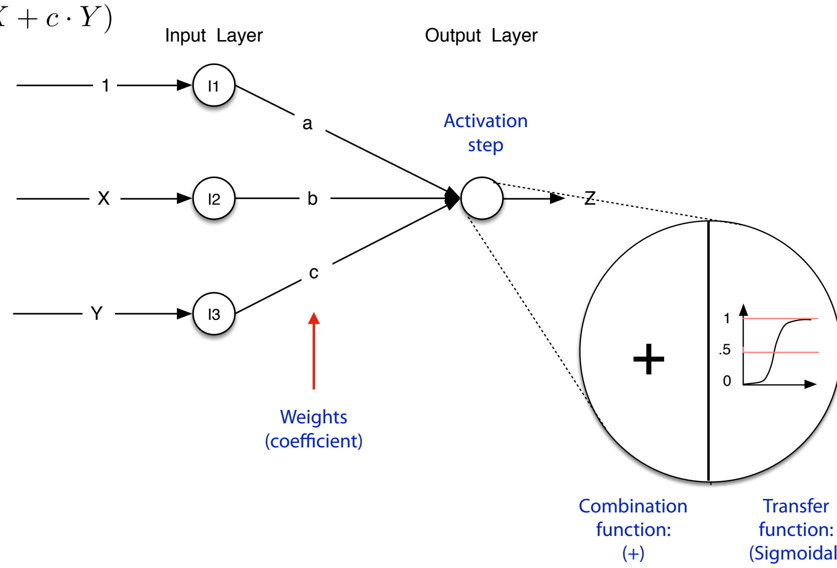
$$Z = a \cdot 1 + b \cdot X + c \cdot Y$$



More typically, neural networks use non-linear functions of inputs

NEURAL NET WITH SIGMOIDAL (S-shaped) TRANSFER FUNCTION, e.g., logistic function

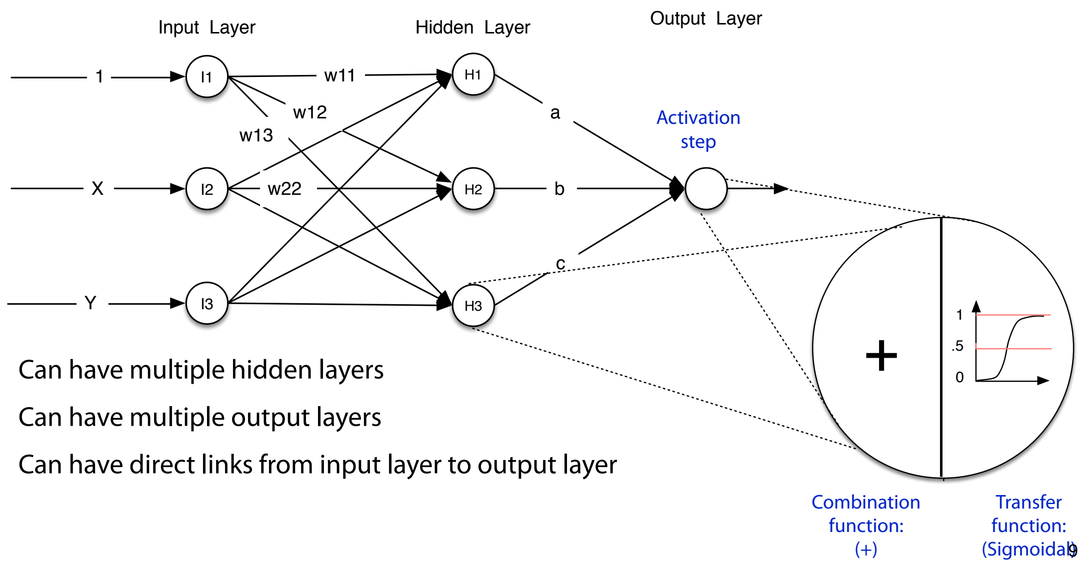
$$Z = \frac{\exp\left(a \cdot 1 + b \cdot X + c \cdot Y\right)}{1 + \exp\left(a \cdot 1 + b \cdot X + c \cdot Y\right)}$$



Neural networks allow for many variations

"STANDARD" NEURAL NETWORK

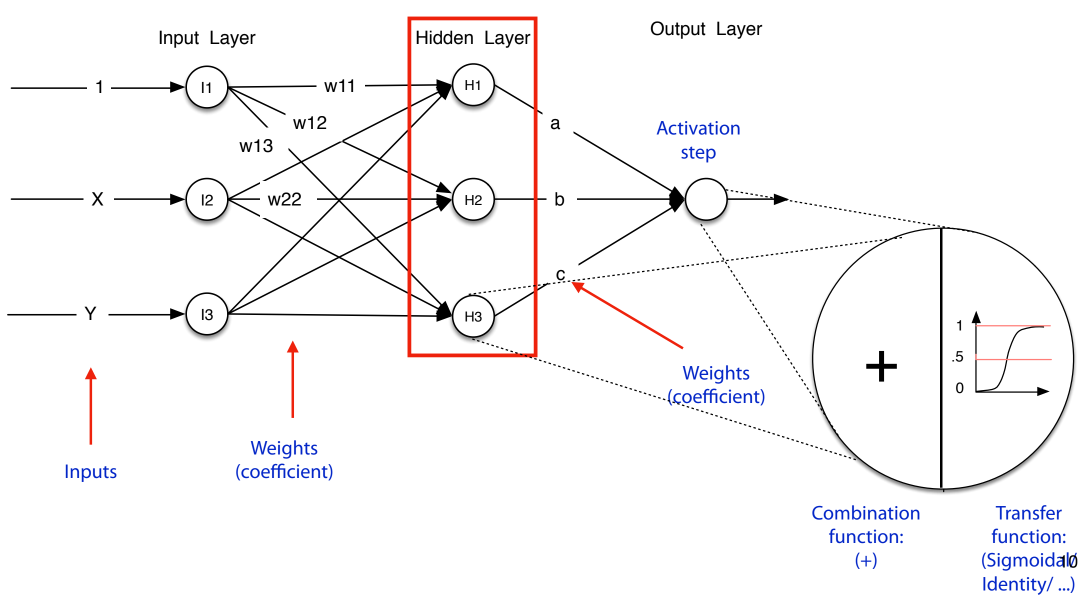
- "Single (multi) layer perceptron"
- "Fully connected, feed forward network with hidden layers and a single node output layer"



- Can have multiple hidden layers
- Can have multiple output layers
- Can have direct links from input layer to output layer

The real power of neural networks comes from applying "interactions" — the hidden layers

NEURAL NET WITH SIGMOIDAL TRANSFER FUNCTION



The Advantage and Pitfall of Neural Net

What do the hidden layers do? High-order Interactions

A heuristic explanation: Taylor Series/Taylor Expansion

$$\frac{1}{1-x} = \sum_{n=0}^{\infty} x^n = 1 + x + x^2 + x^3 + \cdots \qquad R = 1$$

$$e^x = \sum_{n=0}^{\infty} \frac{x^n}{n!} = 1 + \frac{x}{1!} + \frac{x^2}{2!} + \frac{x^3}{3!} + \cdots \qquad R = \infty$$

$$\sin x = \sum_{n=0}^{\infty} (-1)^n \frac{x^{2n+1}}{(2n+1)!} = x - \frac{x^3}{3!} + \frac{x^5}{5!} - \frac{x^7}{7!} + \cdots \qquad R = \infty$$

$$\cos x = \sum_{n=0}^{\infty} (-1)^n \frac{x^{2n}}{(2n)!} = 1 - \frac{x^2}{2!} + \frac{x^4}{4!} - \frac{x^6}{6!} + \cdots \qquad R = \infty$$

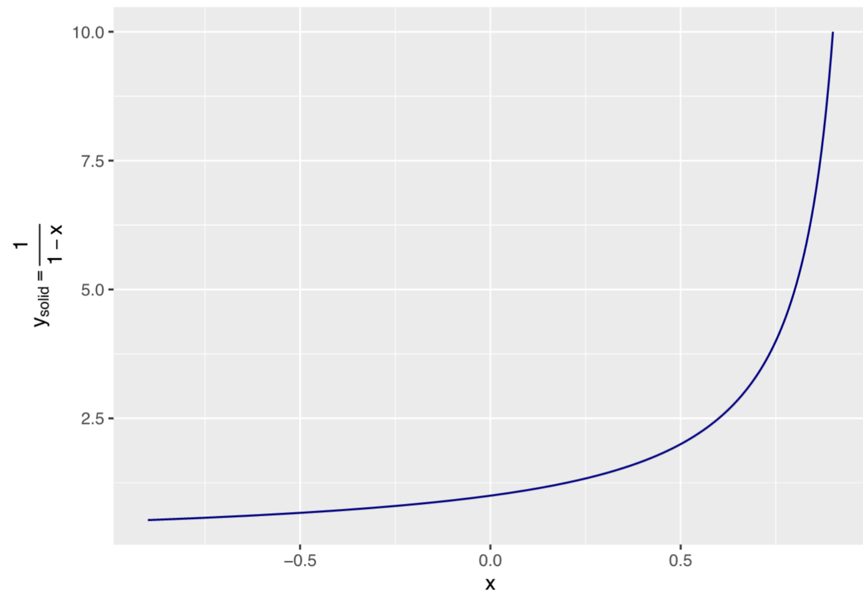
$$\tan^{-1} x = \sum_{n=0}^{\infty} (-1)^n \frac{x^{2n+1}}{2n+1} = x - \frac{x^3}{3} + \frac{x^5}{5} - \frac{x^7}{7} + \cdots \qquad R = 1$$

$$\ln(1+x) = \sum_{n=1}^{\infty} (-1)^{n-1} \frac{x^n}{n} = x - \frac{x^2}{2} + \frac{x^3}{3} - \frac{x^4}{4} + \cdots \qquad R = 1$$

$$(1+x)^k = \sum_{n=0}^{\infty} \binom{k}{n} x^n = 1 + kx + \frac{k(k-1)}{2!}x^2 + \frac{k(k-1)(k-2)}{3!}x^3 + \cdots \qquad R = 1$$

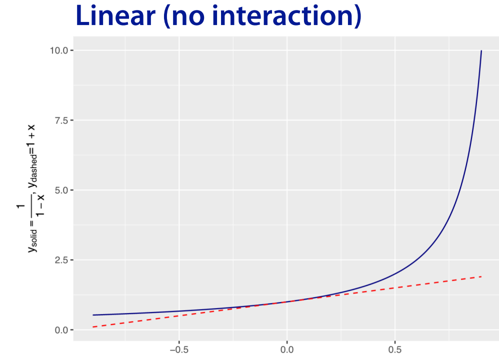
Objective is to predict y based on x

Truth: y=1/(1-x). But we do not know!! Instead, we predict y using x

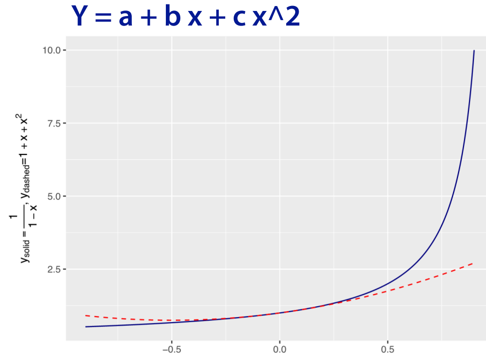


Predict using different levels of interactions

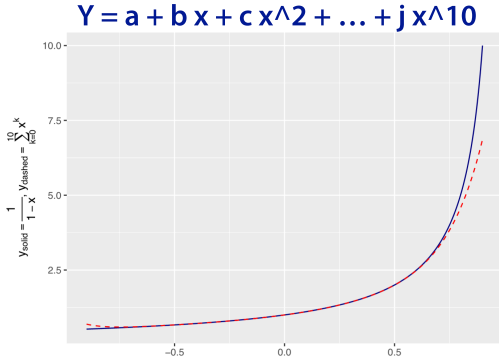
Linear (no interaction)



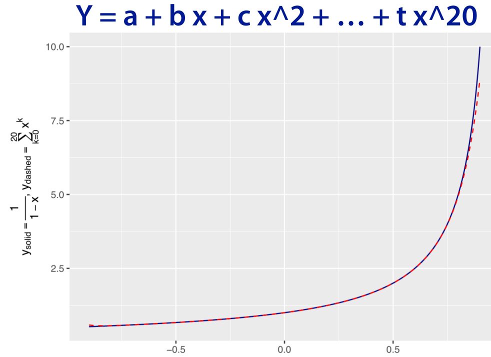
Y = a + b x + c x^2



Y = a + b x + c x^2 + … + j x^10

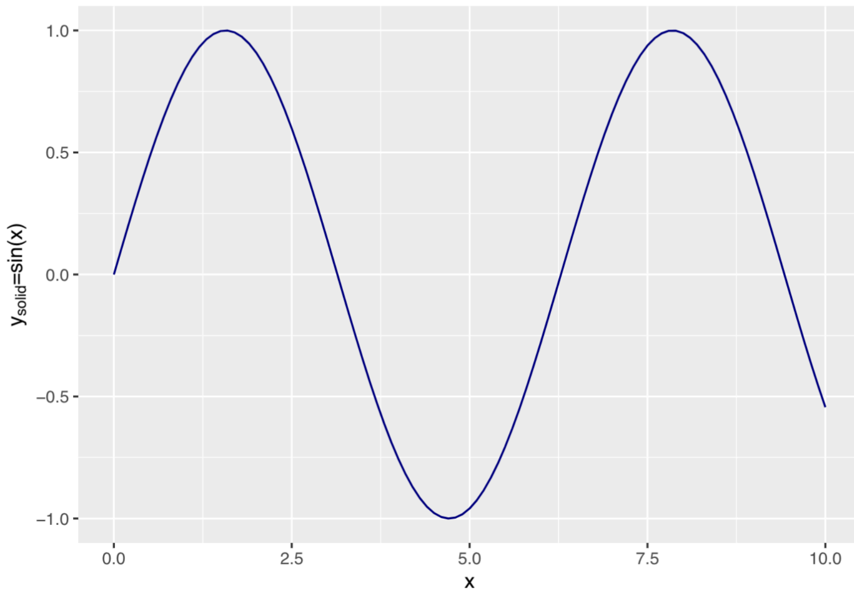


Y = a + b x + c x^2 + … + t x^20



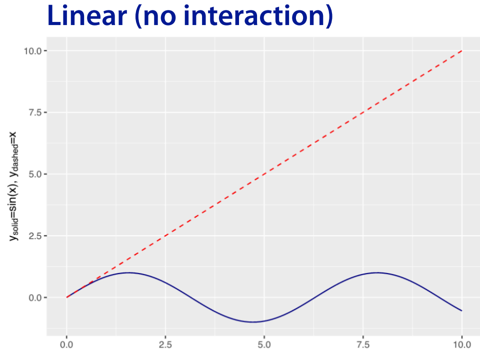
Another example of Taylor series for y=sin(x)

Truth: y=sin(x). But we do not know!! Instead, we predict y using x

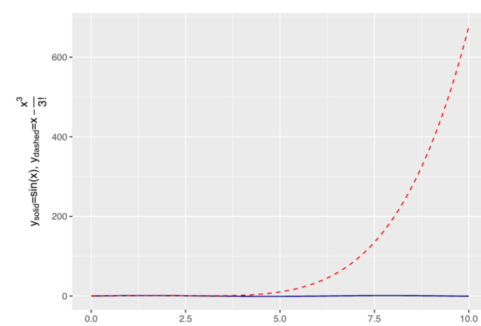


Predict using different levels of interactions

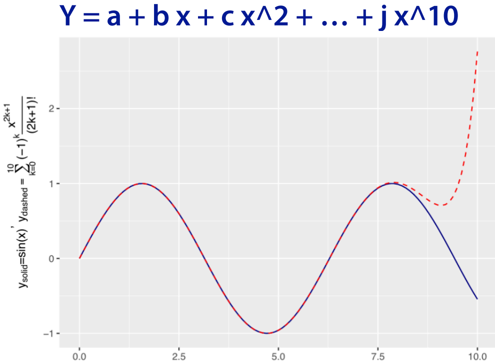
Linear (no interaction)



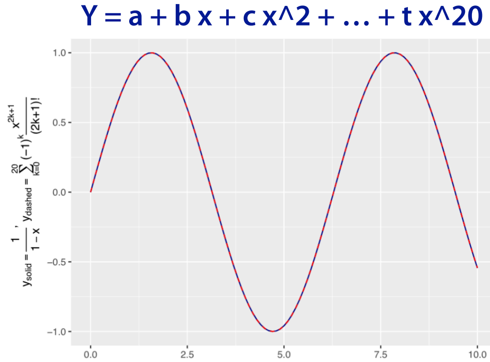
Y = a + b x + c x^2 + d x^3



Y = a + b x + c x^2 + … + j x^10



Y = a + b x + c x^2 + … + t x^20



Neural networks have a key advantage

CONSEQUENCES OF CHOOSING NEURAL NETWORKS (I)

Neural Networks can describe arbitrarily complex data relationships:

- Multilayer perceptrons with sigmoidal (s-shaped) transfer functions are "universal approximators" (the power of the hidden layers!)
- They can theoretically approximate any continuous function to any degree of accuracy
- Nonlinear functions of linear combinations of inputs discover "hidden relationships" for predictions

NNs can be very good at prediction

Neural networks also have two major drawbacks

CONSEQUENCES OF CHOOSING NEURAL NETWORKS (II)

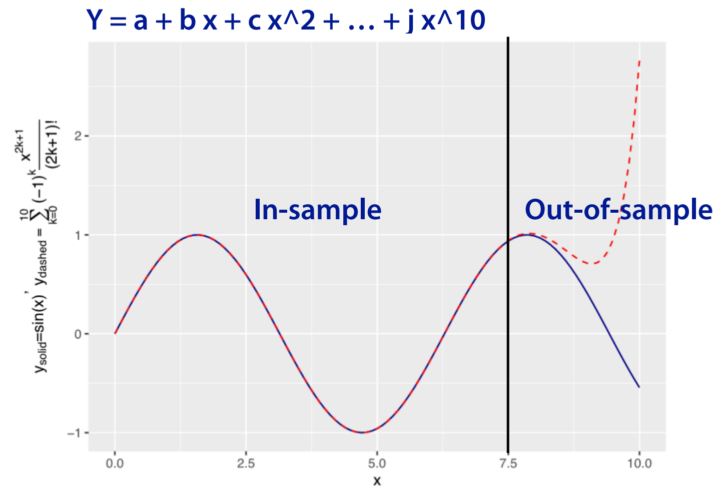
Neural Networks cannot explain results:

- Nonlinear functions of linear combinations of inputs makes interpretation of results nearly impossible
- Problem for business support: Understanding what is going on is often as important as getting good prediction
- Bad when reason is important, e.g. denying loan application

Neural Networks are susceptible to "overfitting":

- **Overfitting:** Can fit well in-sample but badly out-of-sample
- Nonlinear functions of linear combinations of inputs is good at finding "hidden relationships"
  - Good if "hidden relationships" are present in overall customer base
  - Bad if "hidden relationships" are statistical "flukes" of the sample

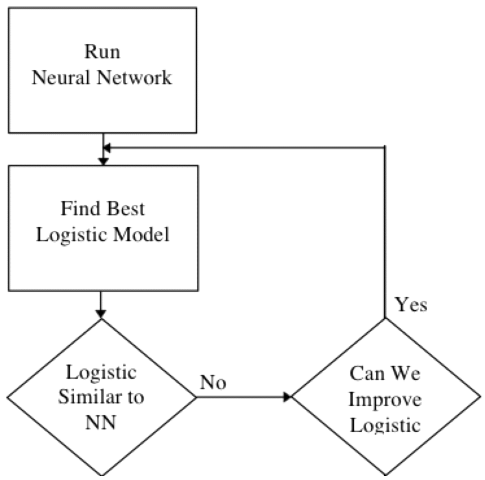
Predict using different levels of interactions



How to take advantage of Neural Net in Customer Analytics?

NNs can be used well in combination with logistic/linear regression for business insights

COMBINING ANN AND LOGISTIC/LINEAR REGRESSION



Core idea:

- Use NN as performance benchmark
- Use logistic/linear regression for variable selection and interpretation

We begin by analyzing the prediction of an NN for whether customers buy "The Art History of Florence"

EXAMPLE 1: NN APPLICATION FOR BOOKBINDERS

- Dave Lawton (marketing director) pulls a random sample of 50,000 customers from the Bookbinders database
- Dave mails "The Art History of Florence" to the entire sample
- 4522 customers buy the book
- Plans to use the NN model to determine which customers to target from the entire database (500,000 remaining customers, excluding test group)

For example we use as input nodes the same variables we used in the logistic regression

EXAMPLE 1: NN APPLICATION FOR BOOKBINDERS

Output Node:

variable name	storage type	display format	value label	variable label
buyer	float	%9.0g	buyer	bought "art history of florence?"

Input Nodes:

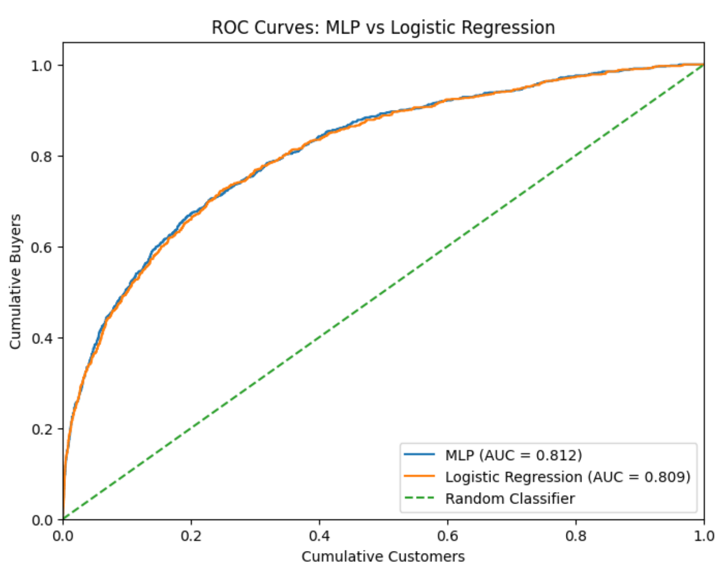
variable name	storage type	display format	value label	variable label
last	float	%9.0g		months since last purchase
total	float	%9.0g		total $ spent
gender	str1	%1s		gender gender
child	float	%9.0g		# purchases, children's books
youth	float	%9.0g		# purchases, youth books
cook	float	%9.0g		# purchases, cookbooks
do_it	float	%9.0g		# purchases, do-it-yourself books
refernce	float	%9.0g		# purchases, reference books
art	float	%9.0g		# purchases, art books
geog	float	%9.0g		# purchases, geography books

Sample:

- 40,000 (randomly selected from the 50,000 random sample)

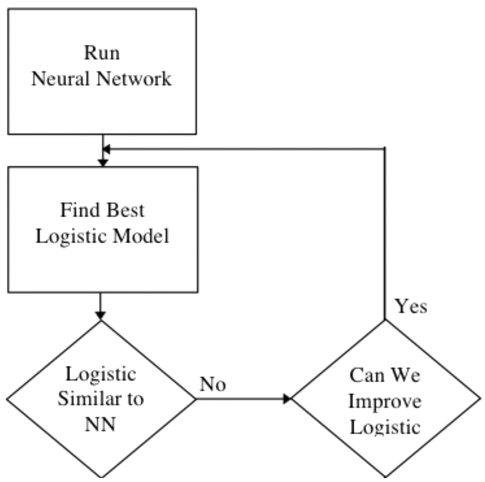
ANN does about as well as logistic model in terms of GAINS (AUC of NN = 0.812; AUC of Logistic = 0.809)

Neural Net:
1. 10-fold Cross Validation
2. Best model has two hidden layers with 5 nodes per layer)
3. AUC shown below is based on the test sample (10,000)



Using the NN gives us confidence that our logistic model is capturing behavior well

COMBINING NN AND LOGISTIC REGRESSION



End Result:

- Confidence in predictive performance
- Easy variable interpretation
  e.g. "art" purchases
  and "male" matter a lot

Consider a second application of NNs

EXAMPLE: FIREWALL WIZARD

- Firewalls on PC are notoriously hard to manage
  (require knowledge of IP ports and networking)
- New "wizard" for configuring Windows firewall
- Profit on firewall wizard is \$10, cost to target customer is \$1
- We have data on 10,000 customers who have been targeted in test campaign
  - Ad-copy A emphasizes "ease of use" (4,607 customers)
  - Ad-copy B emphasizes "control/options" (5,393 customers)
- Available data:

```
res          Is 1 if responded to offer, 0 if not
age          age of customer
numpurch     total number of purchases
totdol       total dollars spent
adB          Is 1 if Ad-copy B 'control/options', 0 if Ad-copy 'A' ease-of-use'
female       Is 1 if female, 0 if male
```

- Goal: Understand the effectiveness of Ad-A and Ad-B

What predicts the response to the firewall offer?

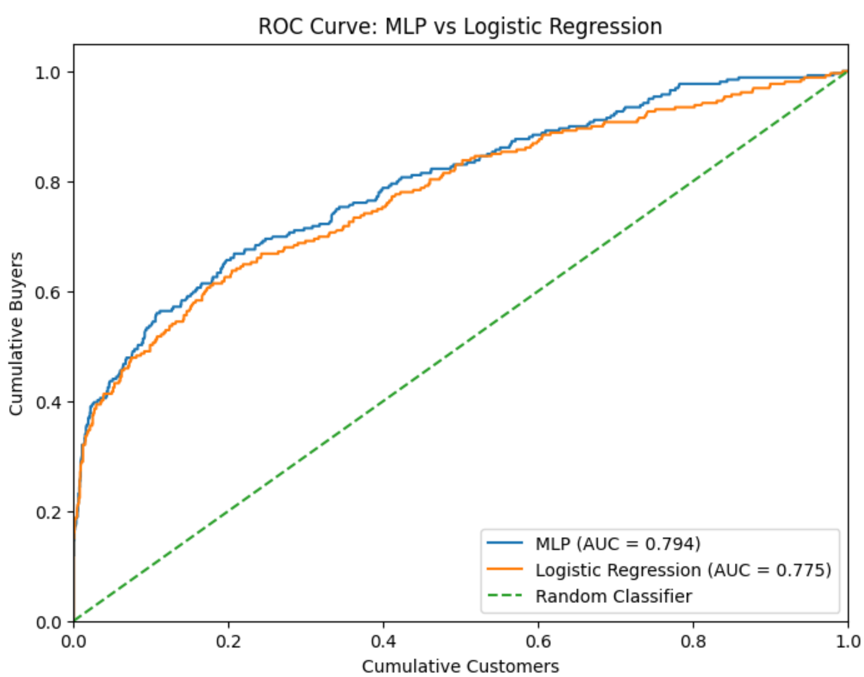
FIREWALL EXAMPLE: RESULTS FROM LOGISTIC REGRESSION

```
                           Logit Regression Results
==============================================================================
Dep. Variable:                    res   No. Observations:                 8000
Model:                          Logit   Df Residuals:                     7994
Method:                           MLE   Df Model:                            5
Date:                Tue, 07 Jan 2025   Pseudo R-squ.:                  0.1805
Time:                        09:06:22   Log-Likelihood:                -2536.3
converged:                       True   LL-Null:                       -3094.9
Covariance Type:            nonrobust   LLR p-value:                 2.527e-239
==============================================================================
                 coef    std err          z      P>|z|      [0.025      0.975]
------------------------------------------------------------------------------
const         -2.0817      0.038    -54.273      0.000      -2.157      -2.007
x1 age         0.2155      0.039      5.597      0.000       0.140       0.291
x2 numpurch    0.2148      0.073      2.934      0.003       0.071       0.358
x3 totdol      0.8647      0.089      9.734      0.000       0.691       1.039
x4 female     -0.0377      0.038     -1.004      0.316      -0.111       0.036
x5 adB         0.0152      0.038      0.406      0.685      -0.058       0.089
==============================================================================
```

- What seems to matter and what not?

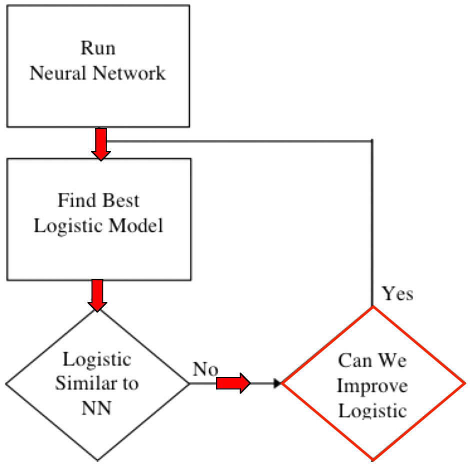
The neural network does substantially better than the logistic regression

GAIN AND COMPARISON FOR NN AND LOGISTIC MODEL



The comparison suggests that the logistic model is missing something important

COMBINING NN AND LOGISTIC REGRESSION



Ideas:

- The effects of Ad A and Ad B may depend on gender!

What predicts the response to the firewall offer?

EXAMPLE 2: RESULTS FROM LOGISTIC REGRESSION w/ Interactions

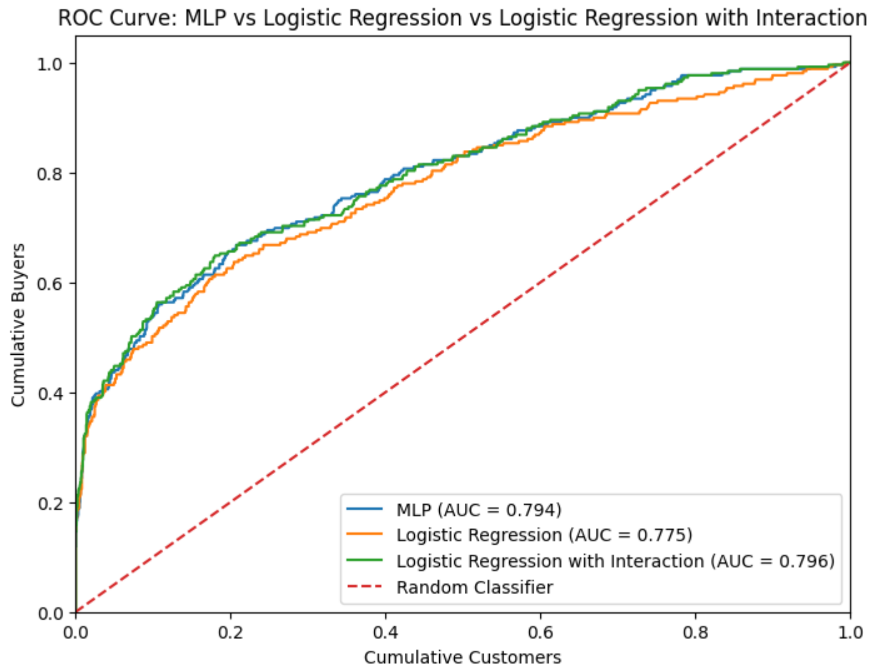
Logit Regression Results

Dep. Variable:	res	No. Observations:	8000
Model:	Logit	Df Residuals:	7993
Method:	MLE	Df Model:	6
Date:	Tue, 07 Jan 2025	Pseudo R-squ.:	0.2056
Time:	09:06:36	Log-Likelihood:	-2458.7
converged:	True	LL-Null:	-3094.9
Covariance Type:	nonrobust	LLR p-value:	1.007e-271

	coef	std err	z	P>|z|	[0.025	0.975]
const	-2.1674	0.041	-52.583	0.000	-2.248	-2.087
x1 age	0.2266	0.039	5.791	0.000	0.150	0.303
x2 numpurch	0.2143	0.074	2.885	0.004	0.069	0.360
x3 totdol	0.9164	0.091	10.122	0.000	0.739	1.094
x4 female	0.4806	0.058	8.240	0.000	0.366	0.595
x5 adB	0.4674	0.056	8.416	0.000	0.359	0.576
x6 female * adB	-0.8477	0.071	-11.977	0.000	-0.986	-0.709

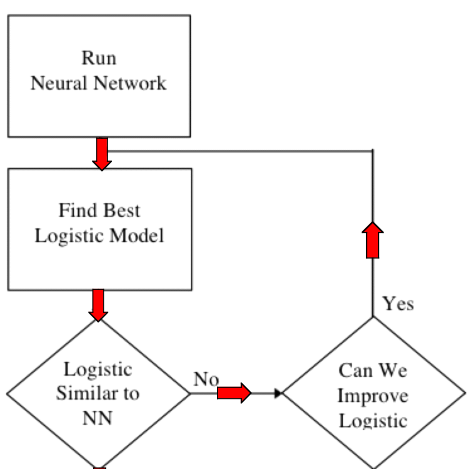
The logistic regression now does as well as the neural network

GAIN AND COMPARISON FOR NN AND LOGISTIC MODELS



The iterative process has revealed something important about consumer behavior

COMBINING NN AND LOGISTIC REGRESSION



End result:

- Confidence in predictive performance
- Easy variable interpretation
- We have learned that the ad-copy appealing to "ease-of-use" appeals more to women and the ad-copy appealing to "control/options" appeals more to men ==> use for ad targeting from now on!

How to detect/mitigate overfitting?

The same characteristic of NNs that helped us to make the logistic model better may lead to “overfitting”

OVERFITTING PROBLEM

- Highly nonlinear structure makes finds "hidden relationships"
- Good if "hidden relationships" are present in overall customer base
- Bad if "hidden relationships" are statistical "flukes" of the sample

==> Can fit well in training sample but badly in test sample = "overfitting problem"

Always split sample into "training" and “testing" sample

We use the Bookbinders case to demonstrate “training” vs. “testing”

TRAINING VS. TESTING PROCEDURE

- Split data ***randomly*** into a “training” set and a “testing” set
    - Normally 70-30 or 80-20 split
- "Train" (calibrate, estimate, fit) the model on the training sample
    - Cross-validation
- Use the trained model to predict using the testing set
- Evaluate model performance based on the testing set