

Introduction to Data Science

Lecture 06; May 4th, 2015

Ernst Henle

ErnstHe@UW.edu

Skype: ernst.predixion

Agenda



- Social Interactions
 - Get and provide help through the LinkedIn group
 - Encourage Group Homework
- Announcements and Midpoint Evaluation
- Review Homework
- Quiz 06a on Schema for Supervised Learning
- Overfitting and Confusion Matrix
- Break
- Probability Threshold, Confusion Matrix, and the ROC chart
- Quiz 06b on ROC and Confusion Matrix
- NoSQL: Scale Out
- Break
- NoSQL: CAP Theorem

Announcements

- Guest Lecture: May 11th 1-hour by Ben Olsen on “Design Concepts for Visualization”
- Guest Lecture: May 18th 1-hour by Marius Marcu “Business Aspects of Data Science” (Changed back to original date)
- May 25th No Class. Memorial Day
- Upcoming Lectures will focus primarily on persistence:
 - NoSQL: CAP, Hadoop, SPARQL
 - Data structure: Relational Algebra, EAV, Graph data

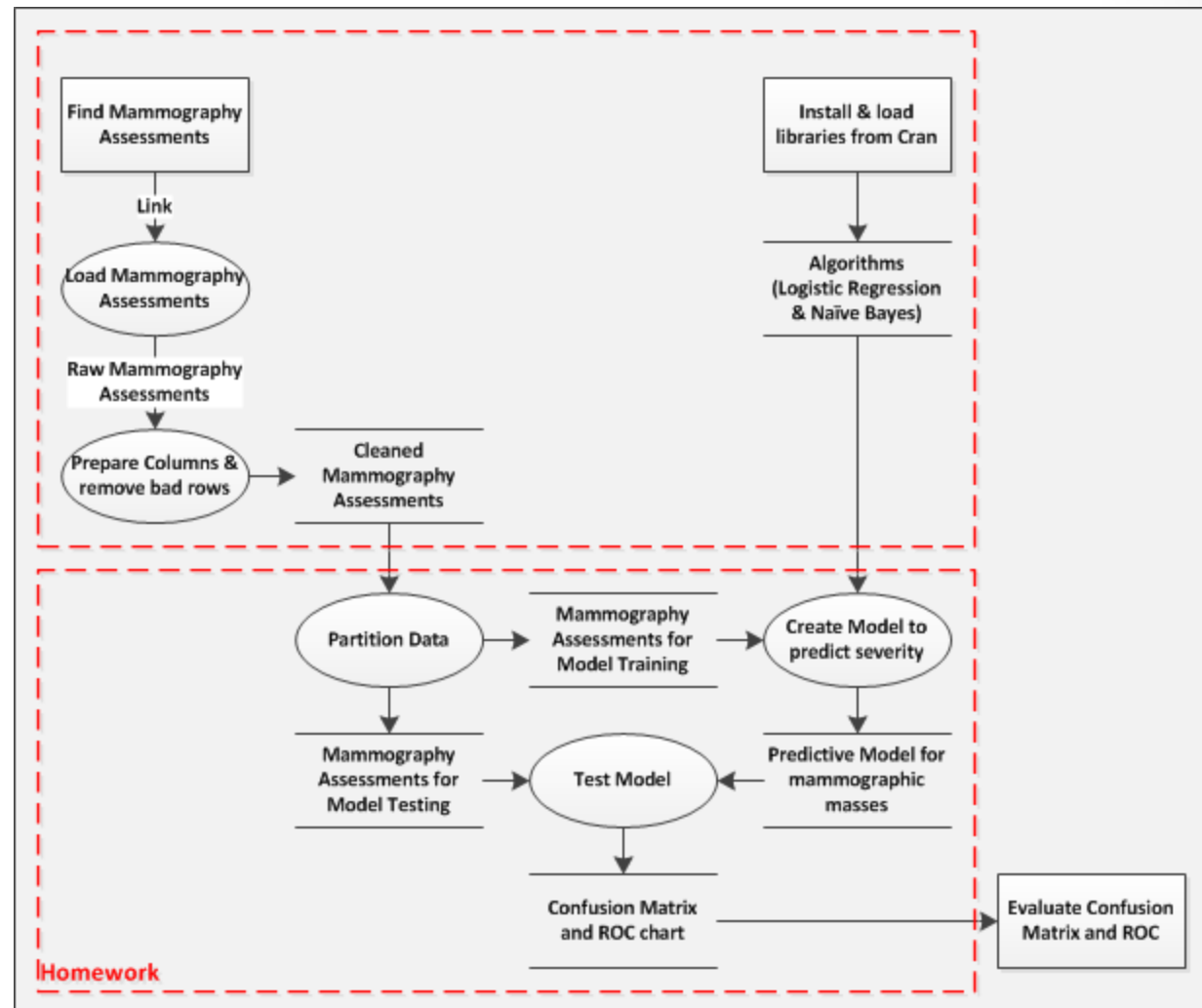
Midpoint Evaluation

- The class has done excellent work. I haven't commented enough on your excellent work and obvious talent.
- Philosophy of instruction: The point of a class like ours, as opposed to a MOOC, is the personal feedback and dedicated community.
 - The feedback from the instructor and students are the most valuable learning tool.
 - The community is special, because everyone is committed to participate.

Midpoint Evaluation

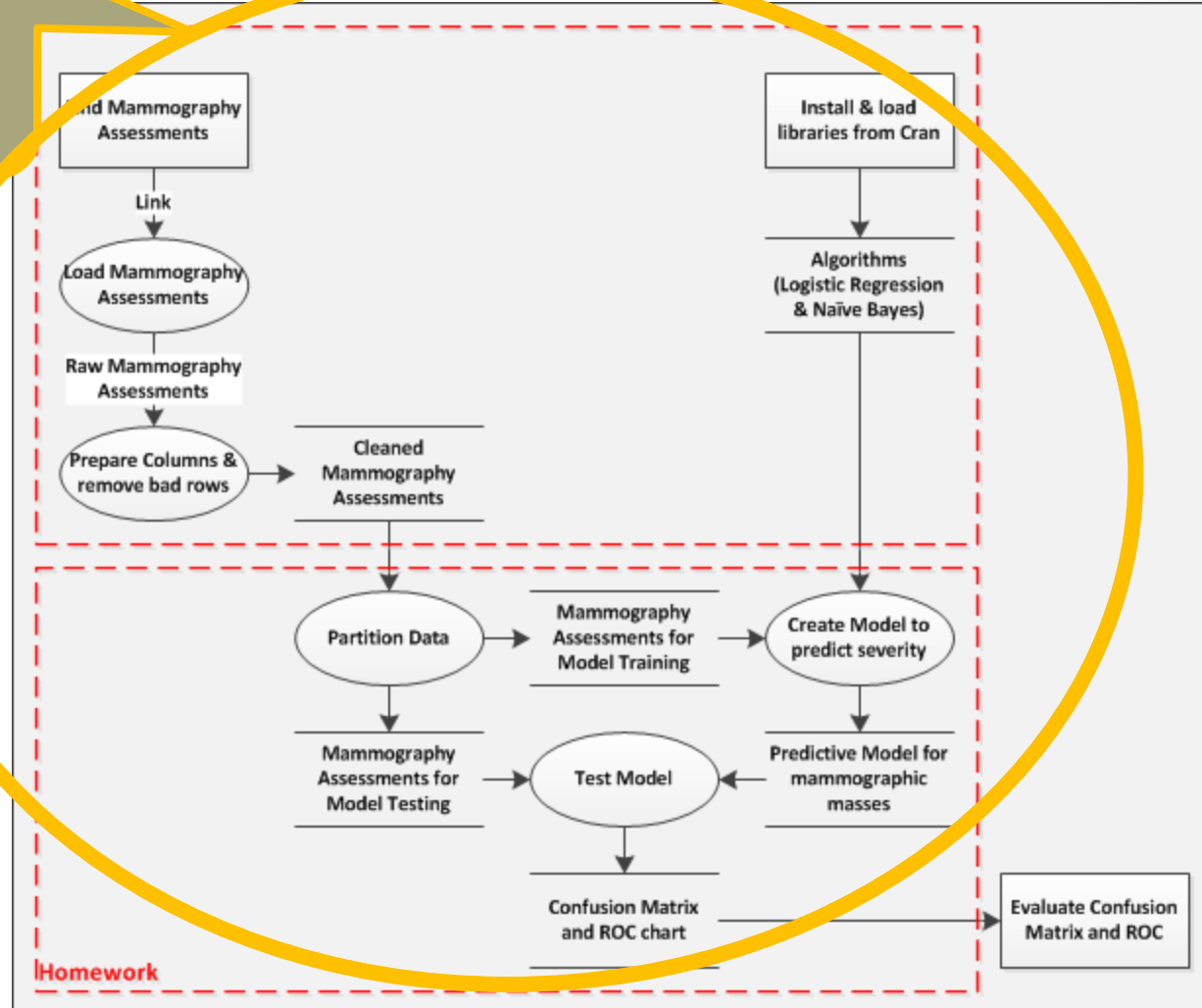
- Some topics we covered (Focus on Analysis)
 - Data Movement: DFD
 - Data Preparation
 - Introduction to MATLAB
 - Introduction to R
 - Predictive Analytics:
 - K-Means
 - Classification
- Some topics we will cover (Focus on Persistence)
 - Fundamental Statistics for Classifications
 - NoSQL (Scale out and CAP)
 - Relational Algebra
 - RDMS
 - Graph Data
 - SPARQL
 - EAV and Sparse Matrices
 - Hadoop (HDFS and MapReduce)

Homework Review: Classifications in R



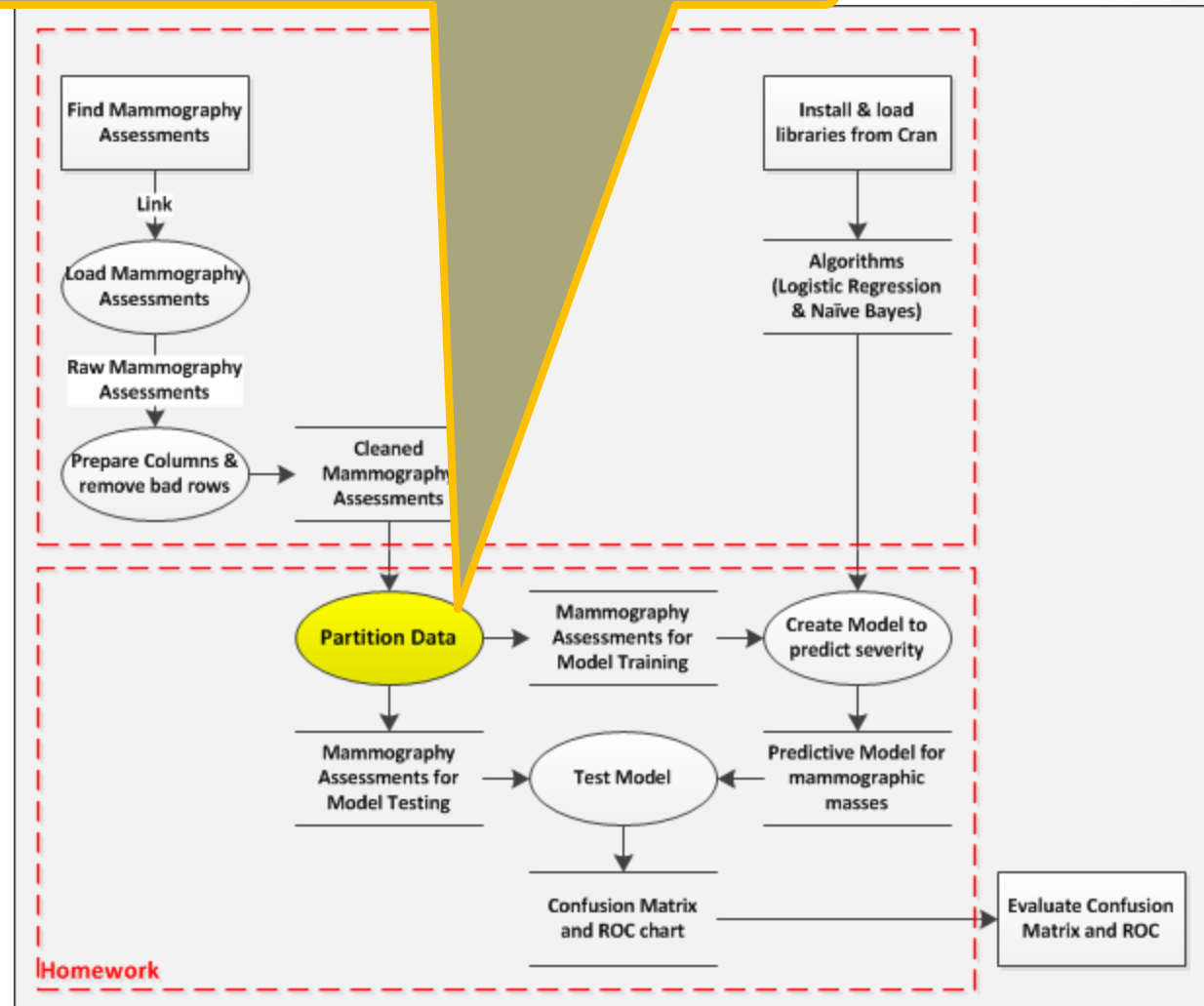
Homework Review: Classifications in R

ClassificationInR.R
&
ClassificationHelper.R



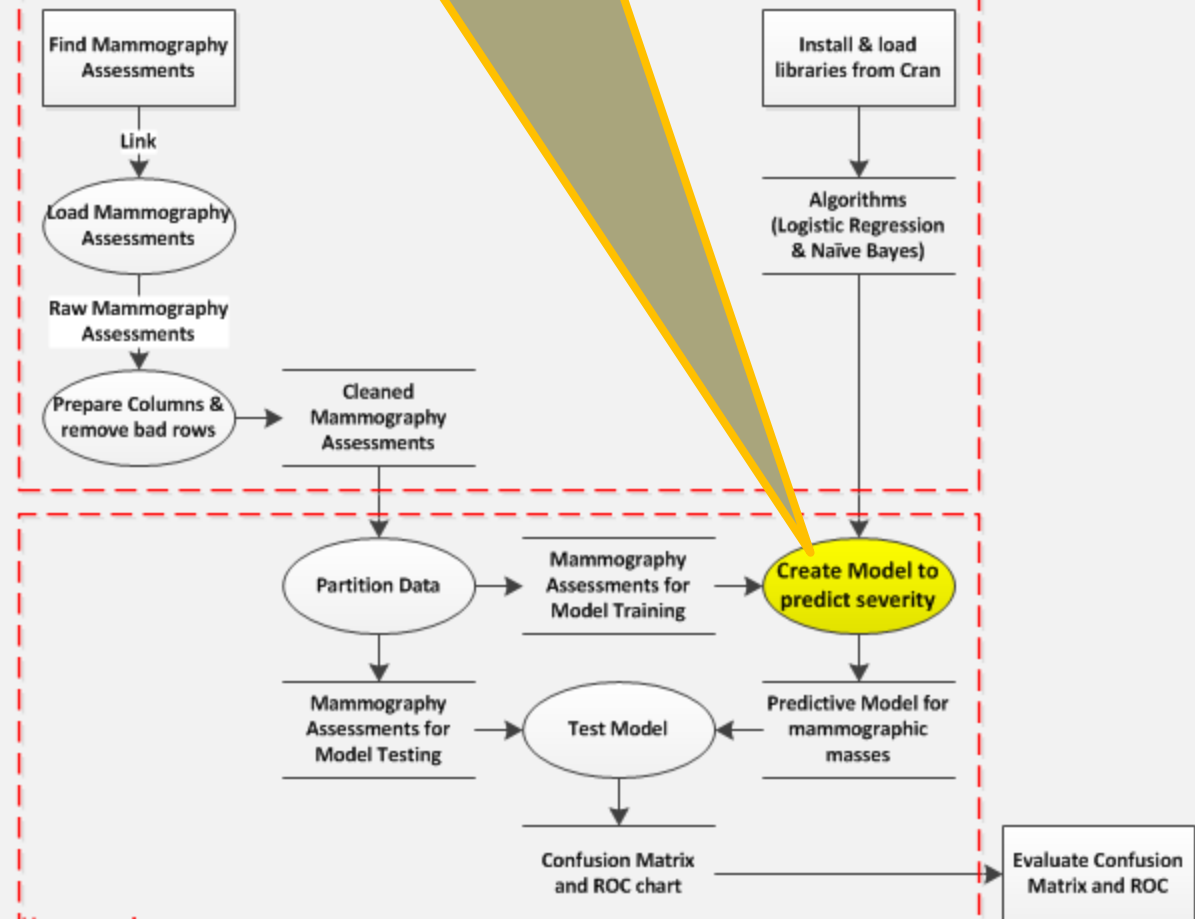
Homework Review: Classifications in R

`FastPartition(dataframe, fractionOfTest=0.4)`



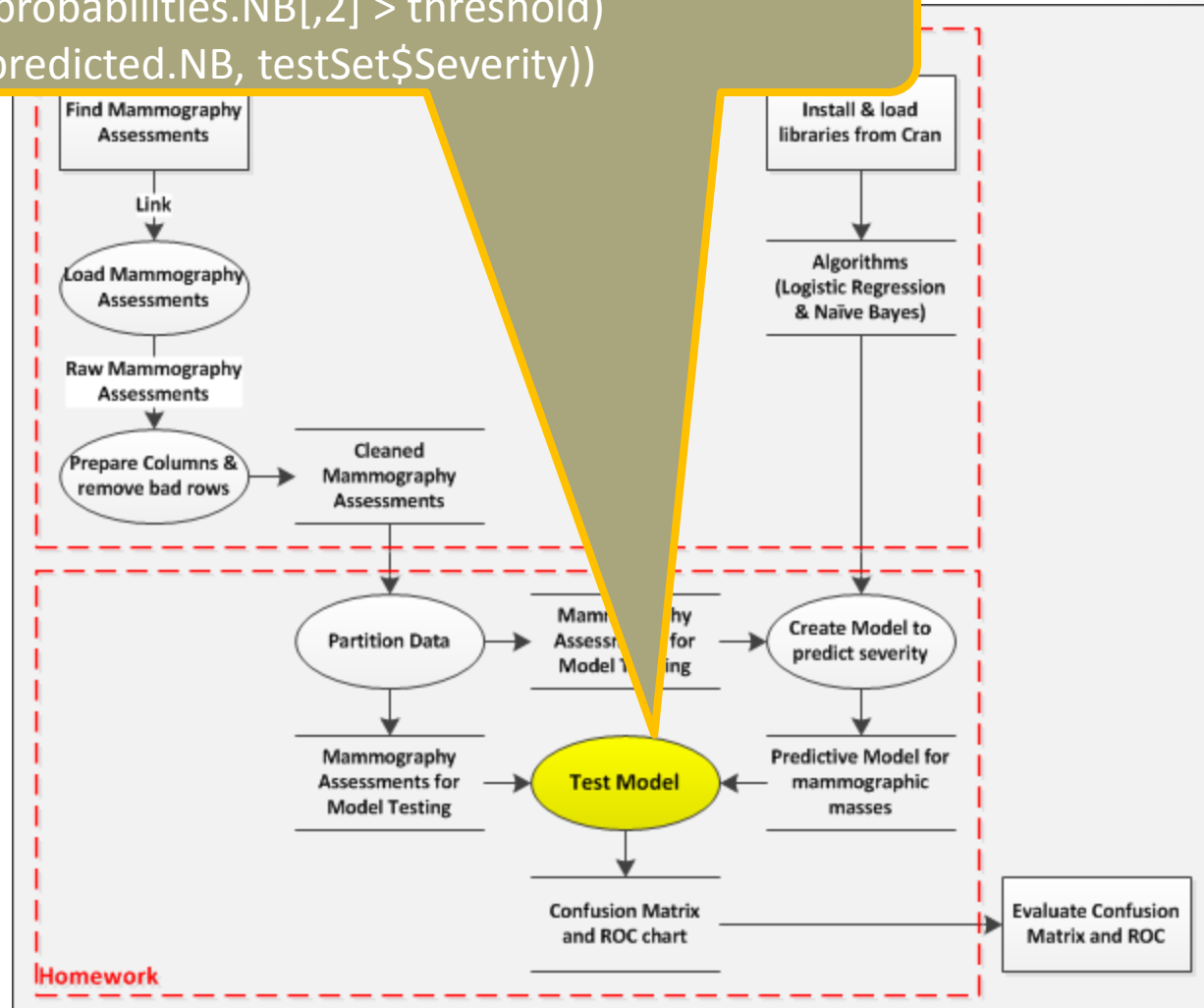
Homework Review: Classifications in R

```
naiveBayes(formula, data=trainSet)
```

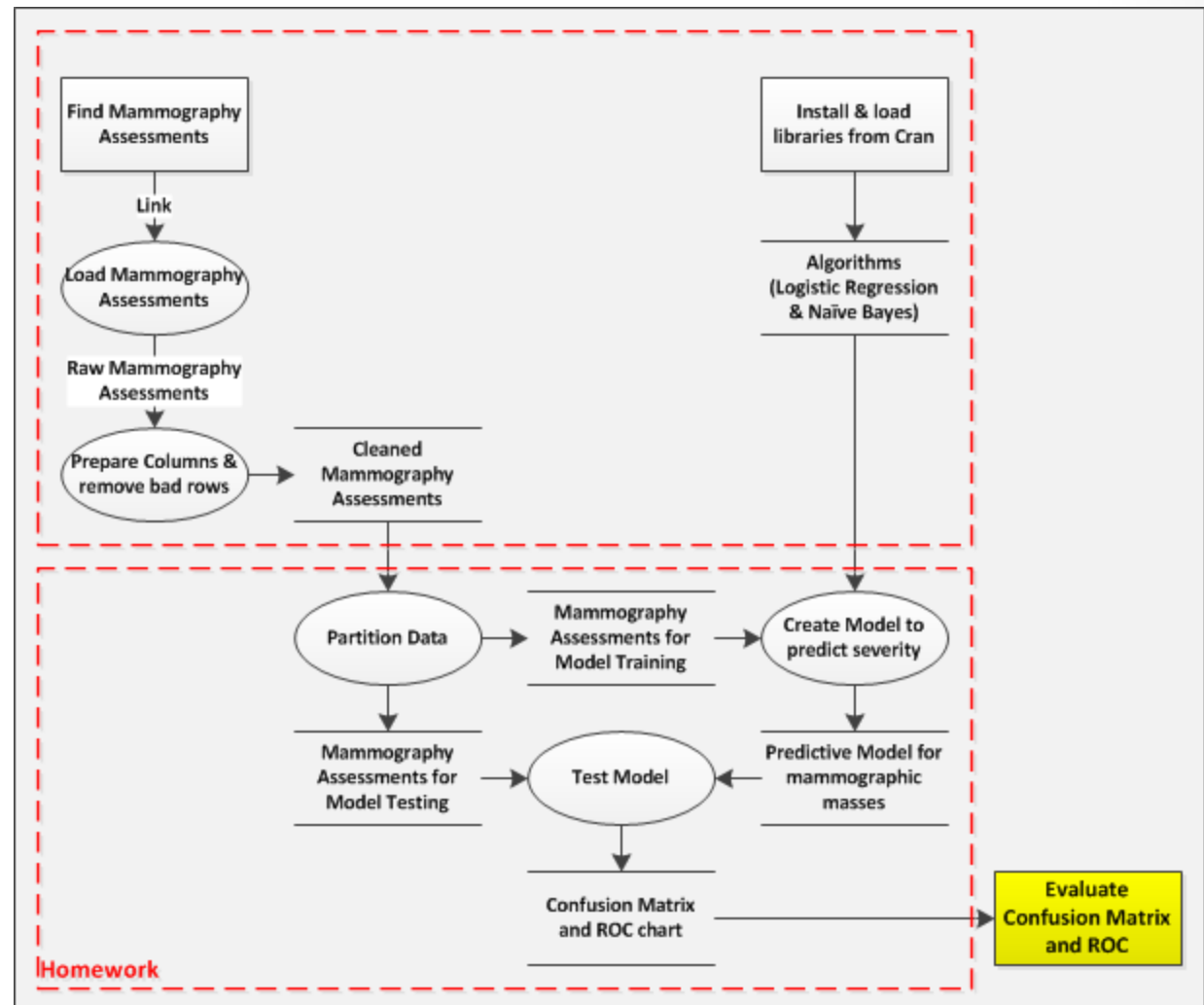


Homework Review: Classifications in R

```
predict(naiveBayesModel, newdata=testSet, type="raw")  
as.numeric(probabilities.NB[,2] > threshold)  
print(table(predicted.NB, testSet$Severity))
```



Homework Review: Classifications in R

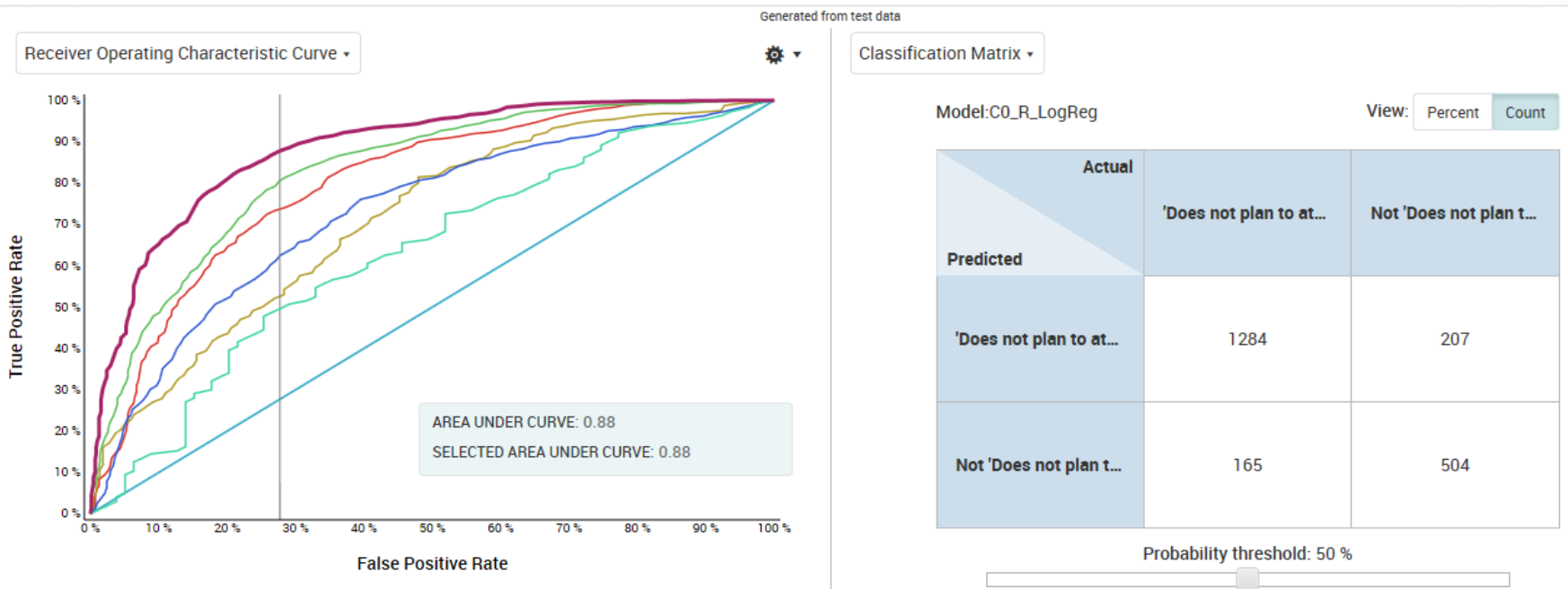


Homework Review

- Classification Demo:

<http://www.predixionsoftware.com/help/webframe.html#01ClassifyWalkthrough.html>

- Confusion Matrix and ROC Chart

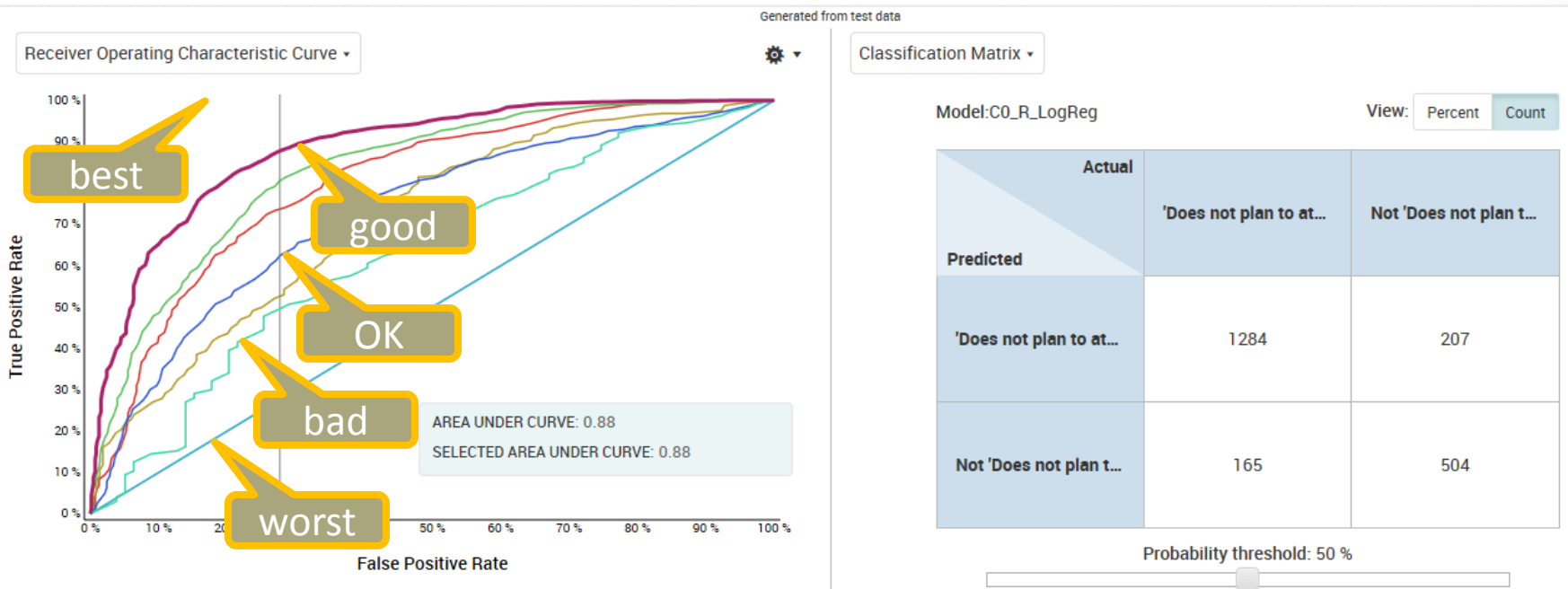


Homework Review

- Classification Demo:

<http://www.predixionsoftware.com/help/webframe.html#01ClassifyWalkthrough.html>

- Confusion Matrix and ROC Chart

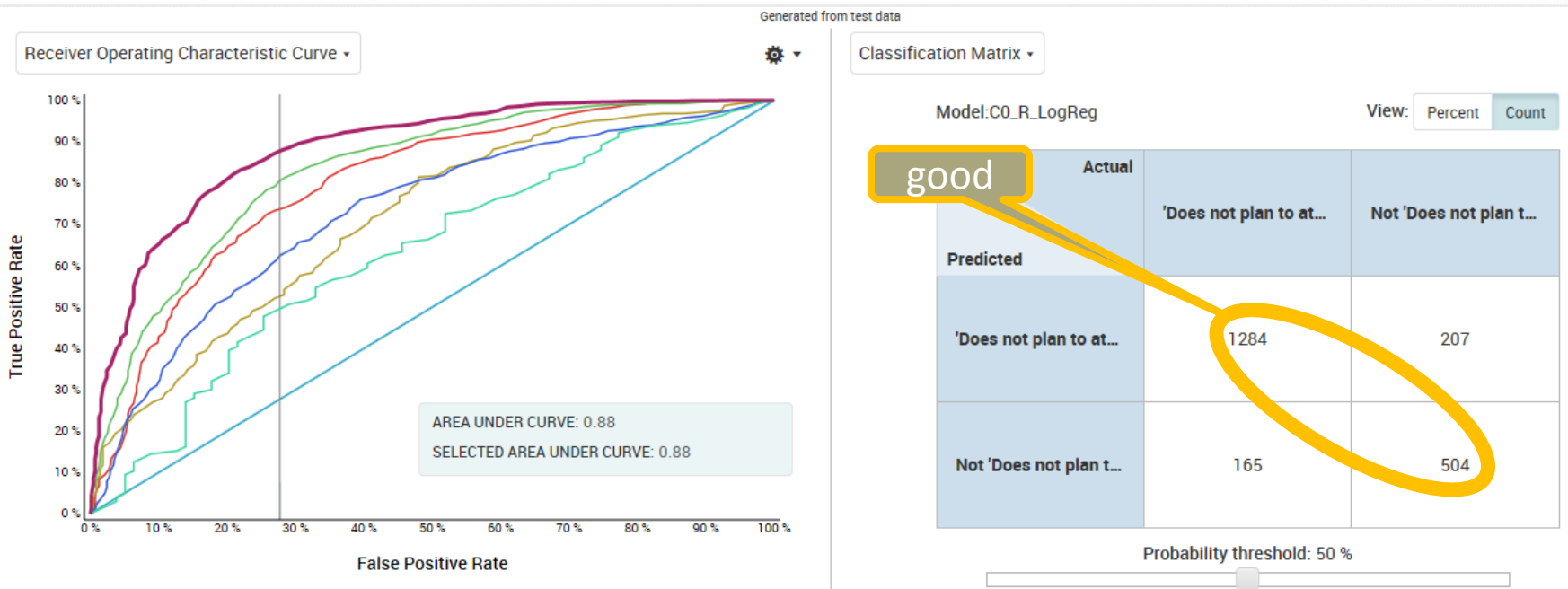


Homework Review

- Classification Demo:

<http://www.predixionsoftware.com/help/webframe.html#01ClassifyWalkthrough.html>

- Confusion Matrix and ROC Chart



Homework Review

- Classification Demo:

<http://www.predixionsoftware.com/help/webframe.html#01ClassifyWalkthrough.html>

- Confusion Matrix and ROC Chart



Quiz 06a

- <https://catalyst.uw.edu/webq/survey/ernsthe/270012>
- Schema and Supervised Learning
- You may need to look up the answer for question 7

Over-fitting and Confusion Matrix

Ernst Henle

ErnstHe@UW.edu

Skype: ernst.predixion

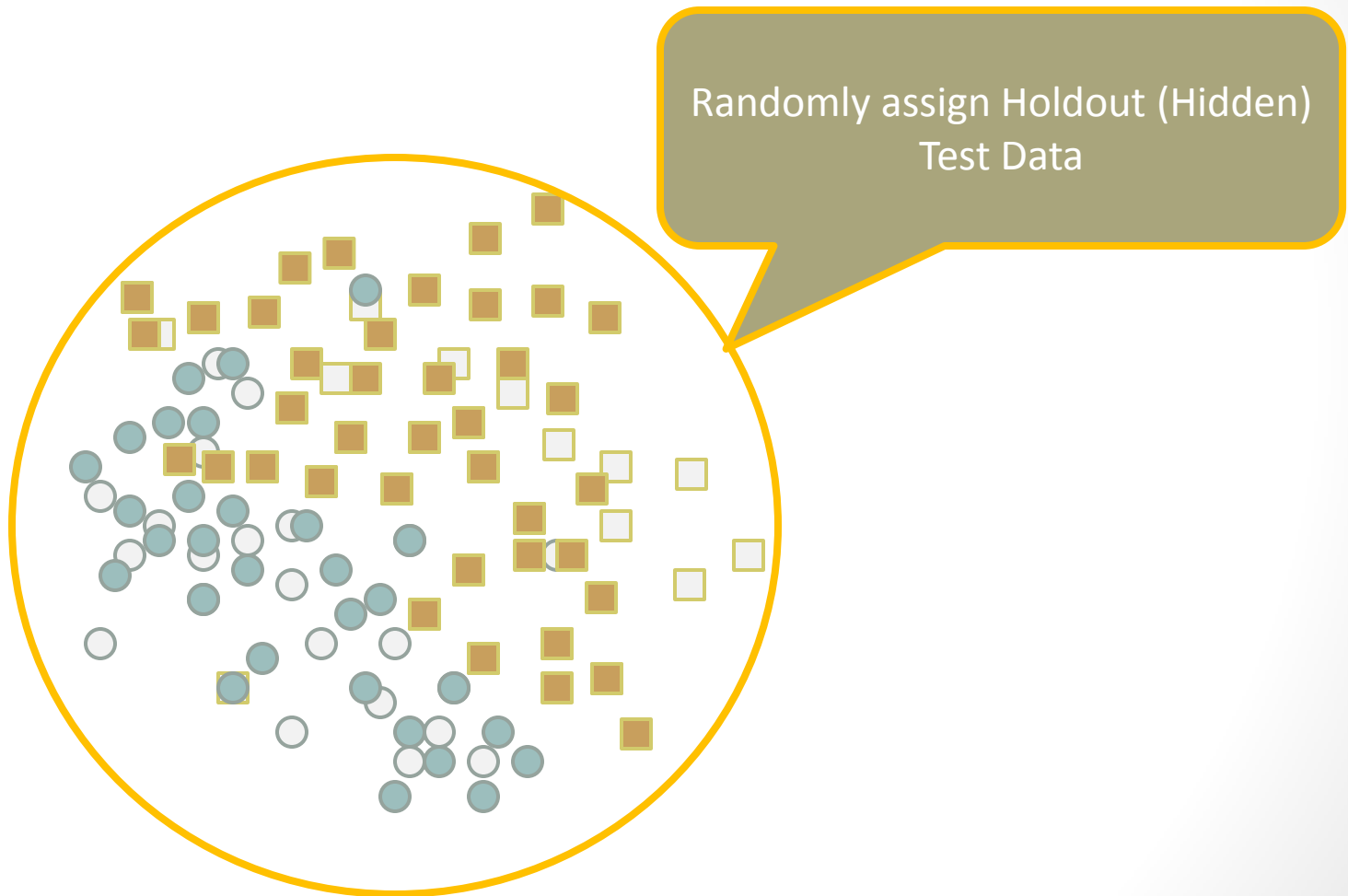
Evaluate Model

- Use an over-fitting example to explain the following concepts:
 - Modeling Data
 - Training Data
 - Test Data
 - Model (Hypothesis)
 - Over-fitting
 - Model Accuracy
 - Confusion Matrix (Classification Matrix)
 - True Positive
 - False Positive
 - True Negative
 - False Negative

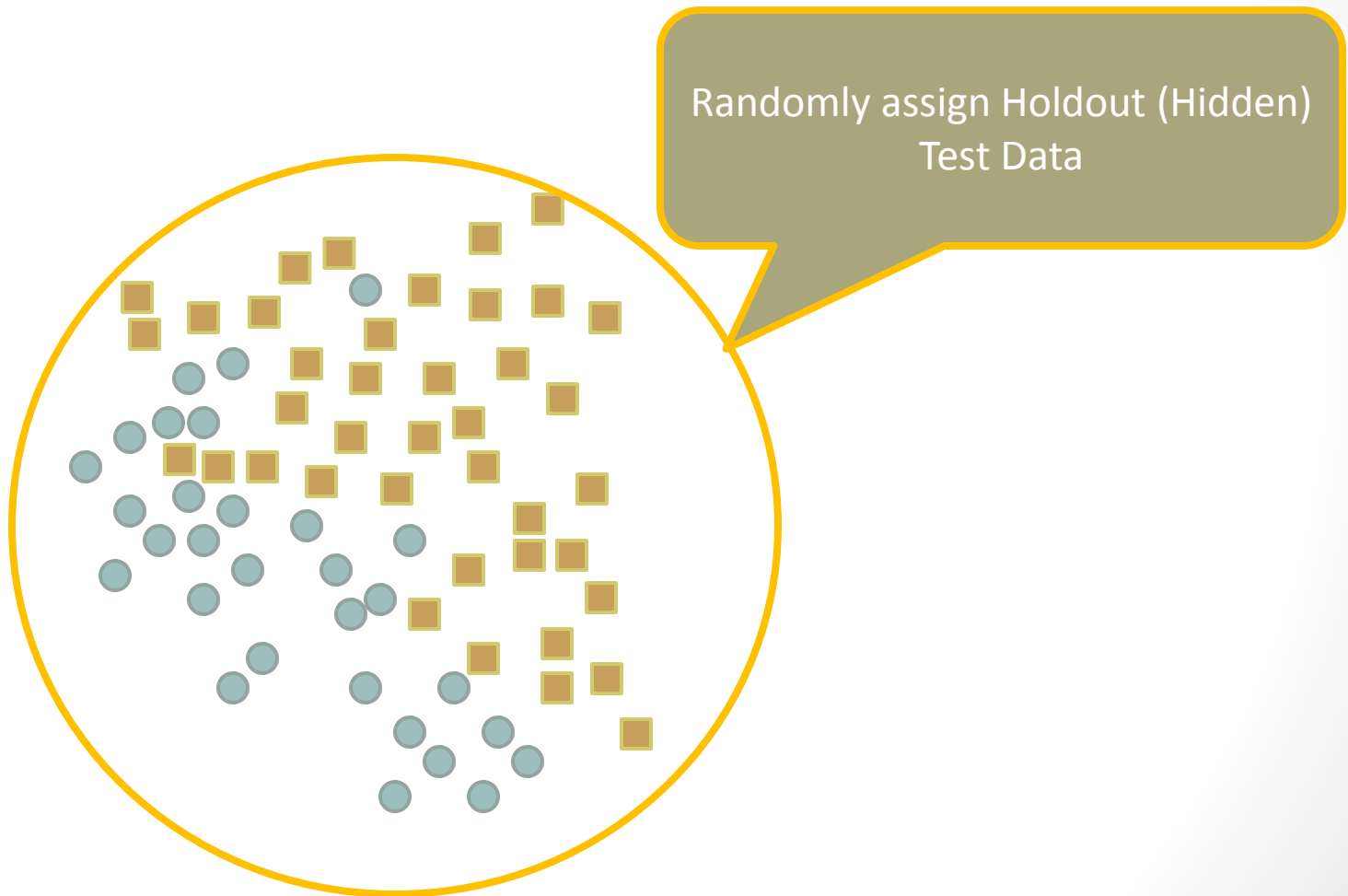
Evaluate Model: All Data



Evaluate Model: Test Data



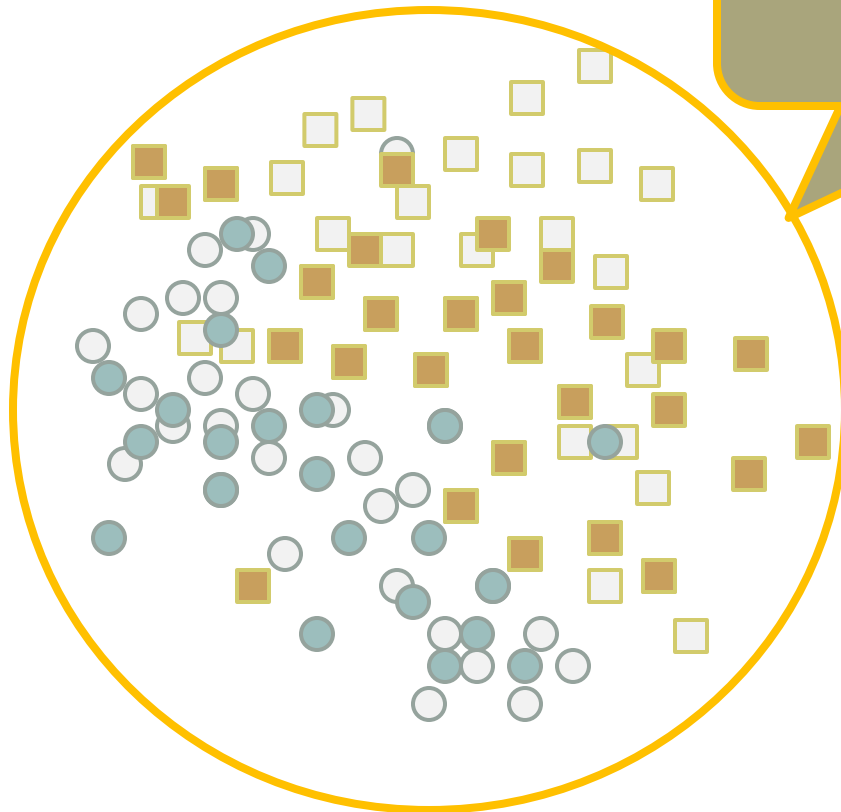
Evaluate Model: Test Data



Evaluate Model: All Data

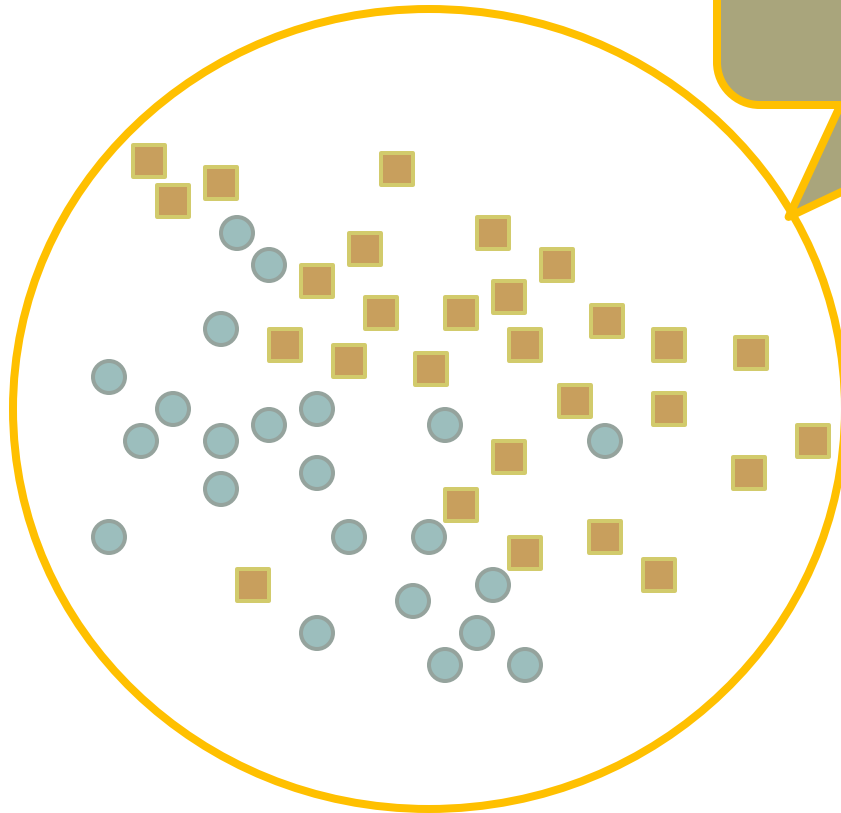


Evaluate Model: Training Data



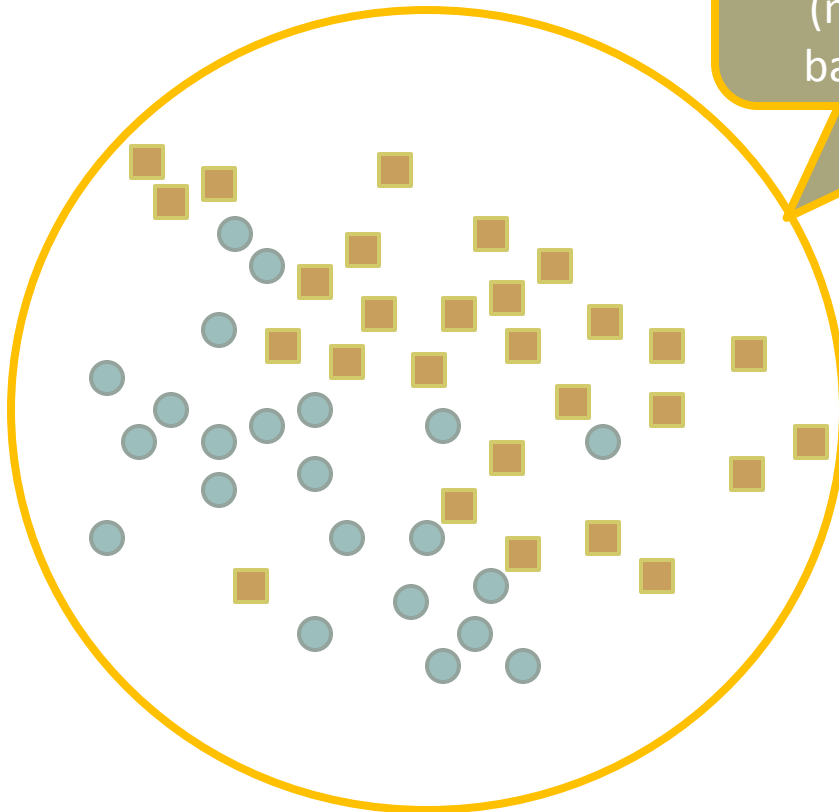
The Data that is not Test Data is
used for Training

Evaluate Model: Training Data



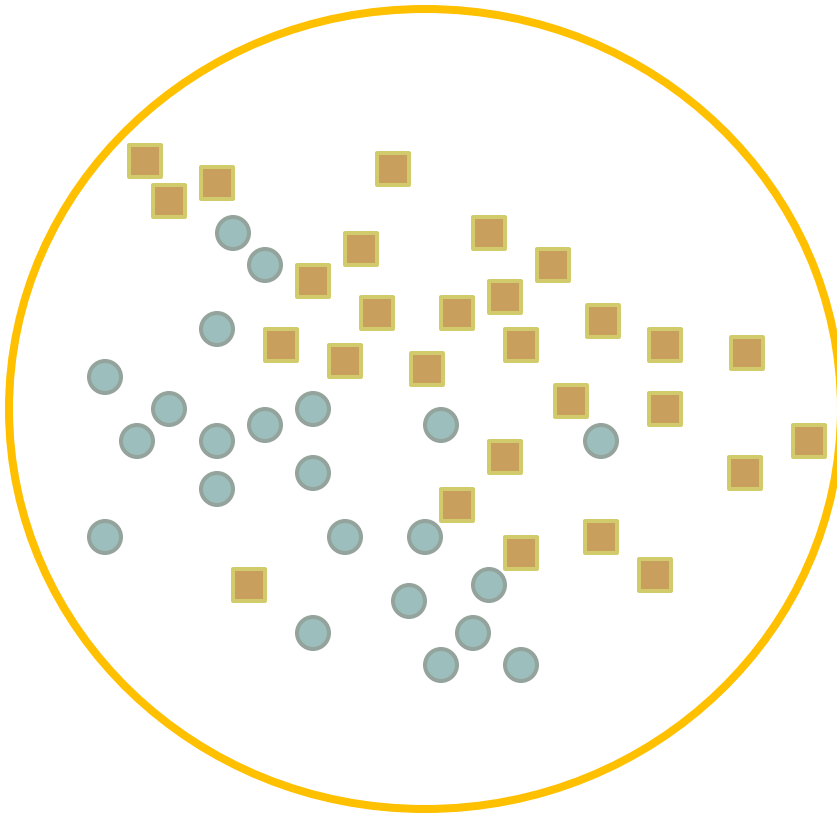
The Data that is not Test Data is
used for Training

Evaluate Model: Training



I want to predict if a point is a square (positive) or a circle (negative). The prediction is based on the point's location

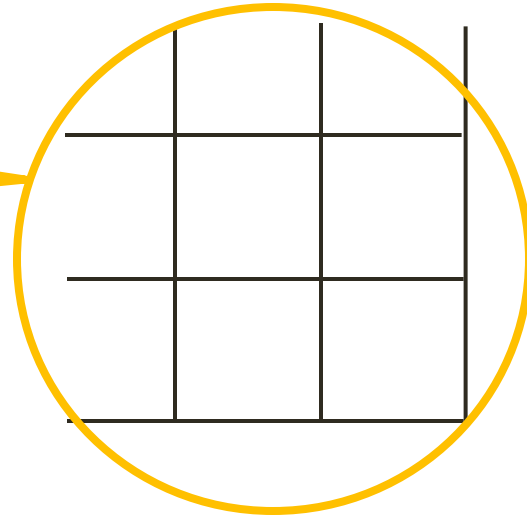
Evaluate Model: Training

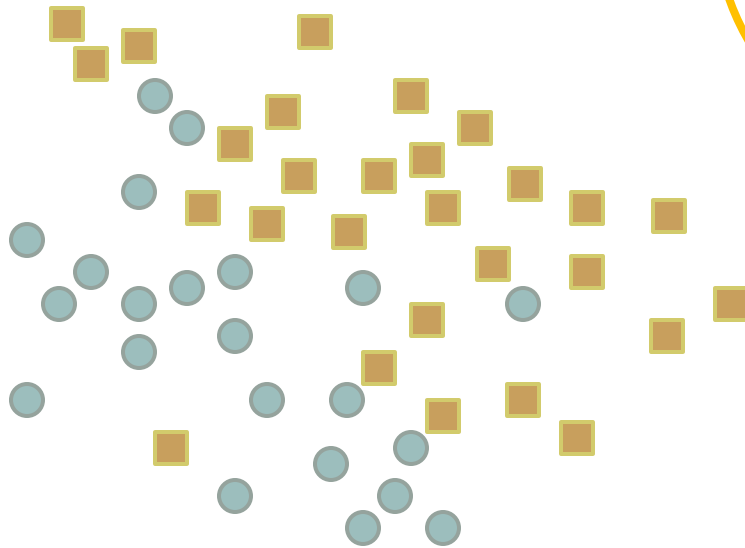


$\text{isSquare} \sim x\text{Location} + y\text{Location}$

Evaluate Model: Confusion Matrix

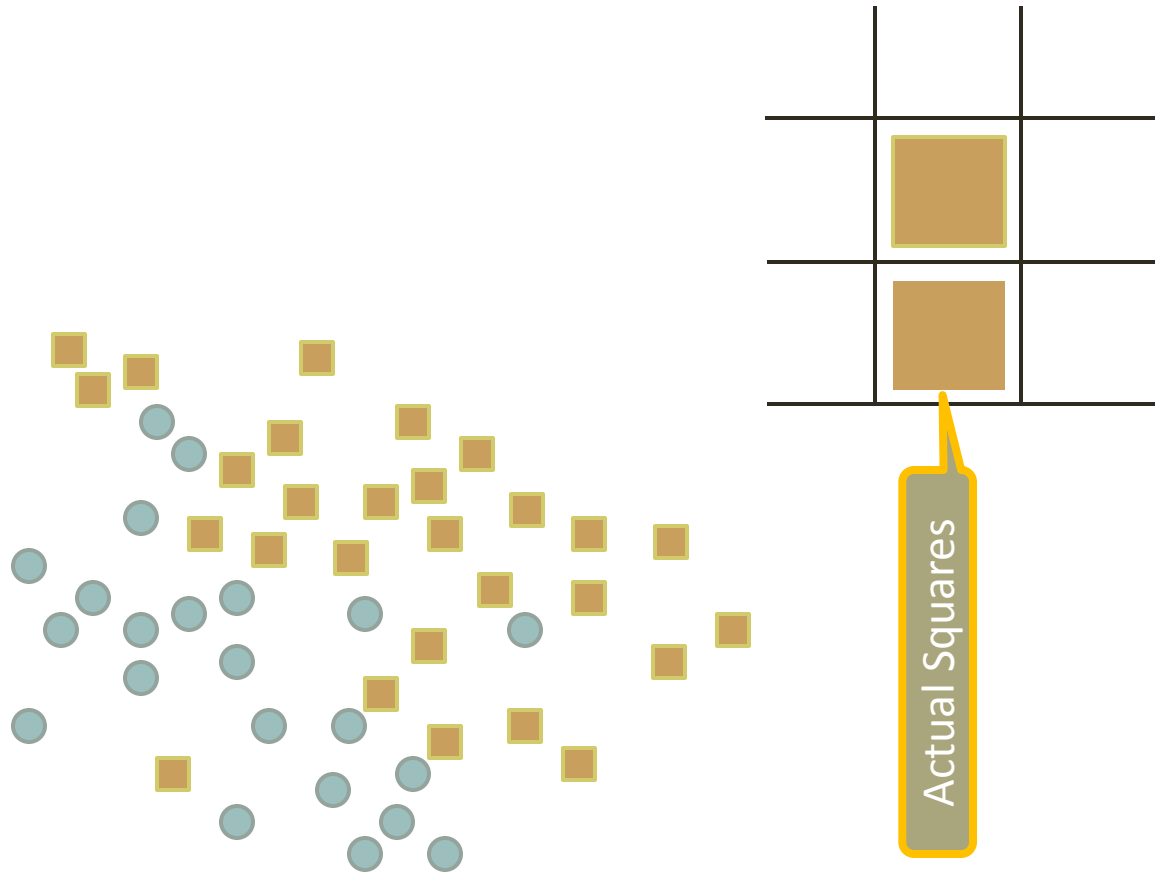
Confusion Matrix (Classification Matrix):
Compare Squares and Circles with
Predicted Squares and Circles





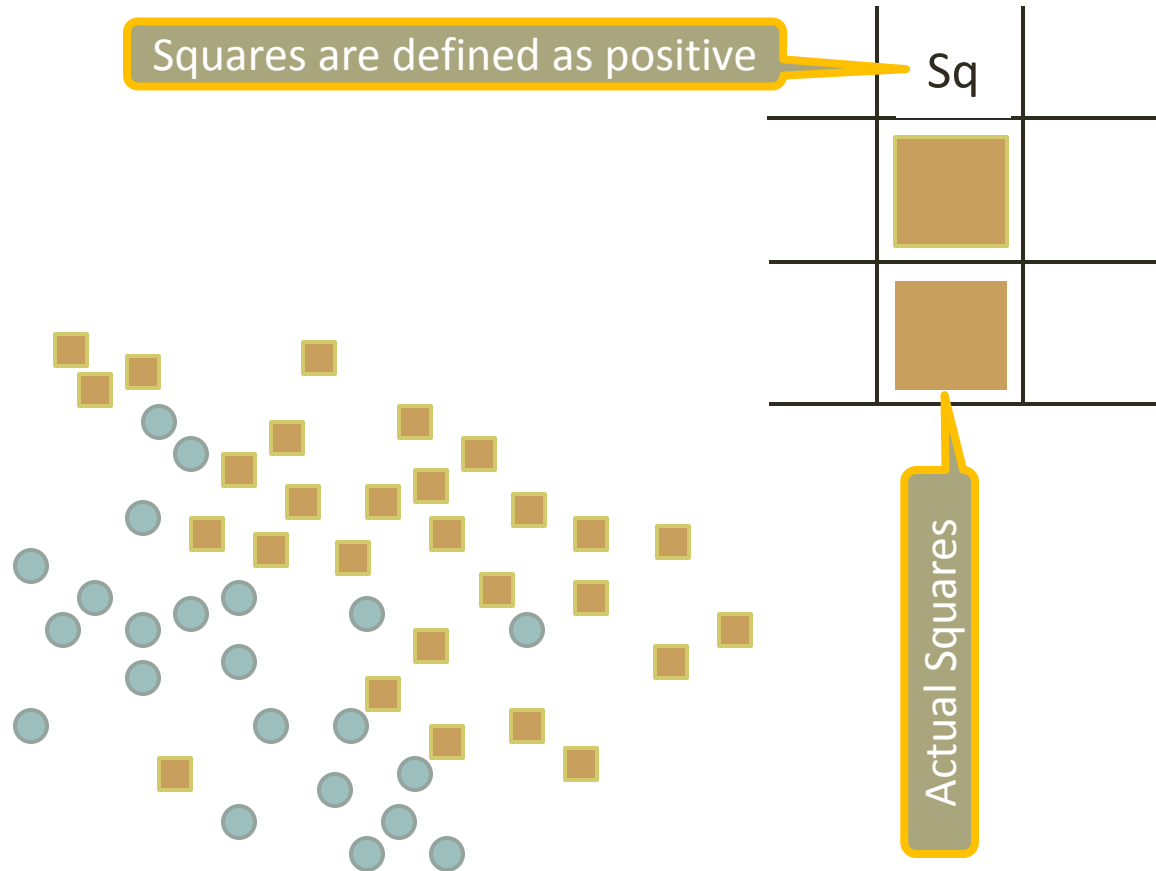
$\text{isSquare} \sim \text{xLocation} + \text{yLocation}$

Evaluate Model: Confusion Matrix



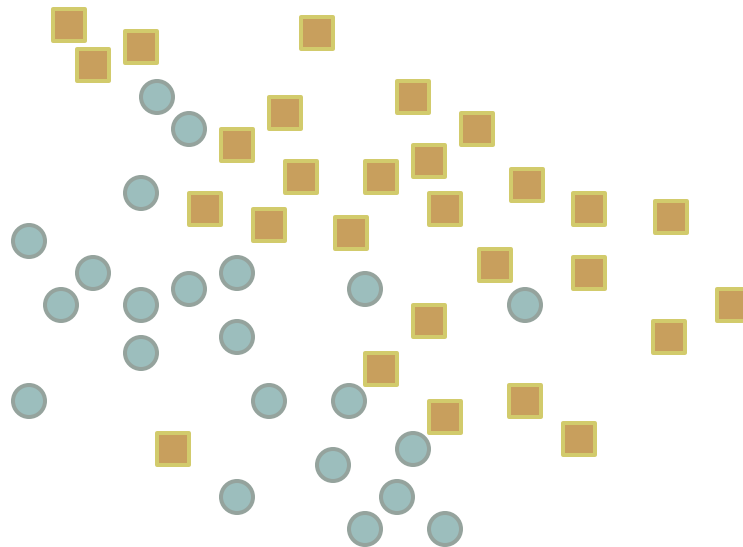
$\text{isSquare} \sim \text{xLocation} + \text{yLocation}$





Evaluate Model: Confusion Matrix



$\text{isSquare} \sim \text{xLocation} + \text{yLocation}$

Evaluate Model: Confusion Matrix

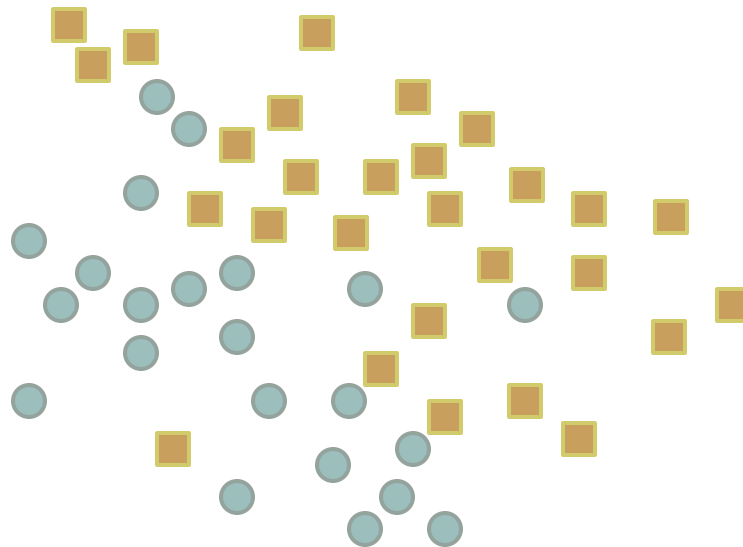


	Sq	
		
		





Actual Circles

$\text{isSquare} \sim \text{xLocation} + \text{yLocation}$

Evaluate Model: Confusion Matrix



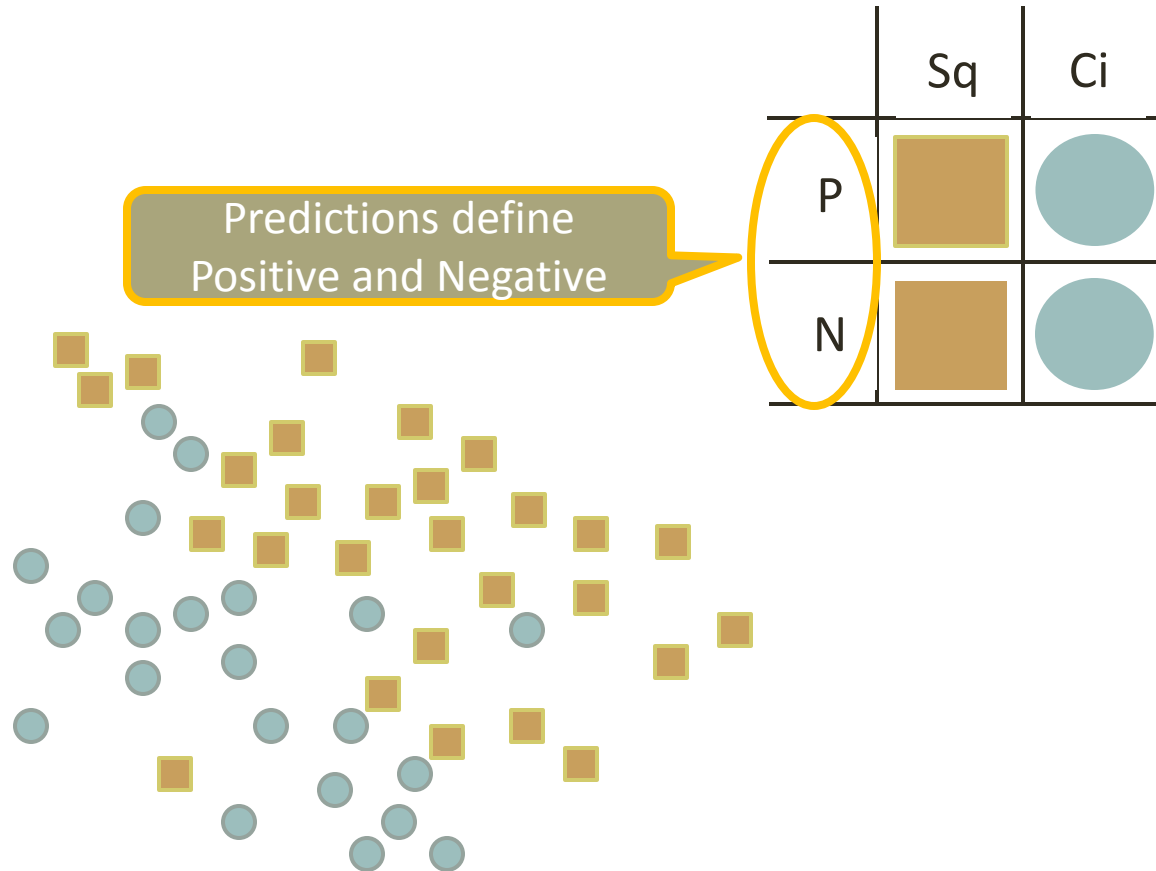
Circles are defined as negative

	Sq	Ci
		
		

Actual Circles

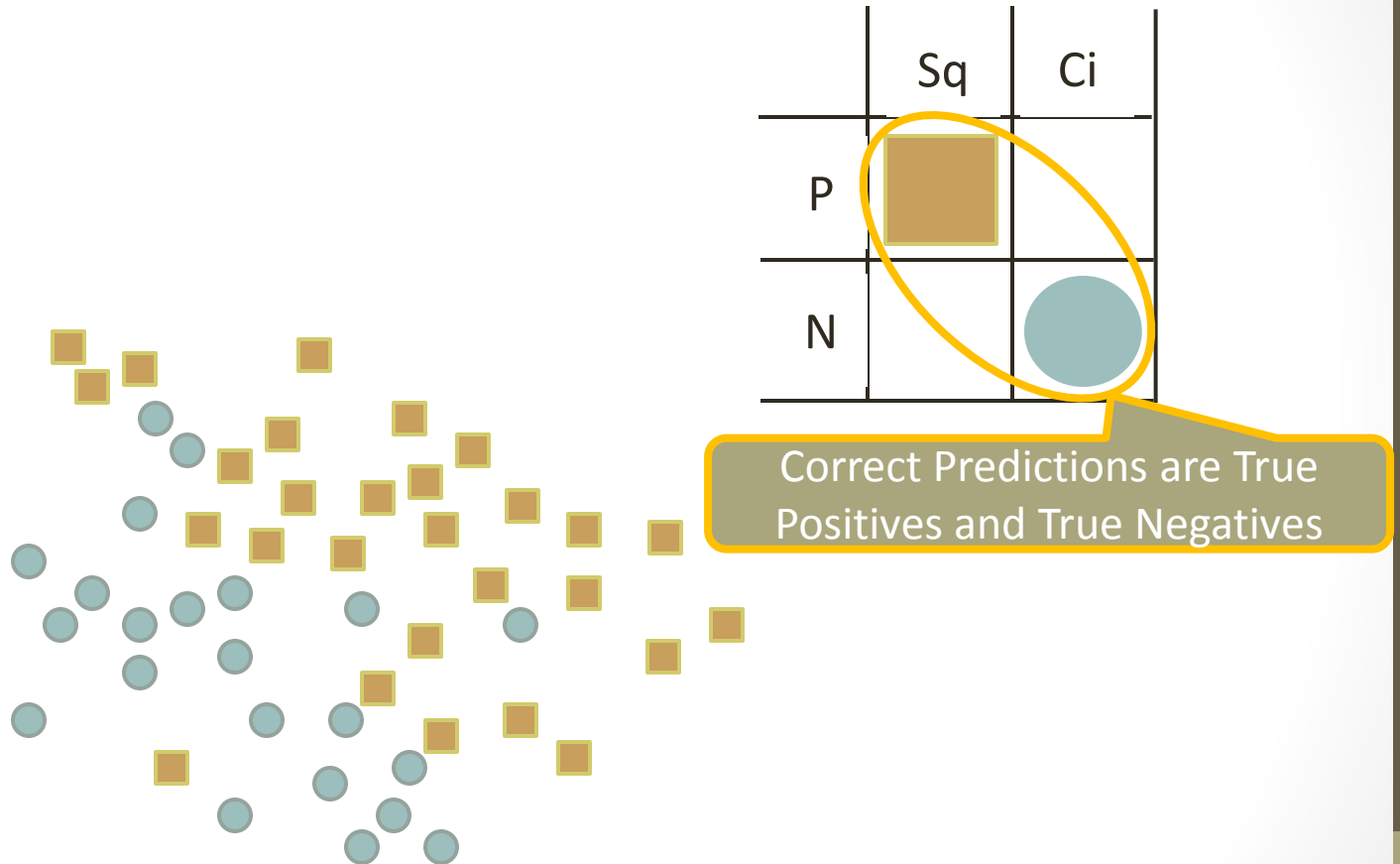
$\text{isSquare} \sim \text{xLocation} + \text{yLocation}$

Evaluate Model: Confusion Matrix



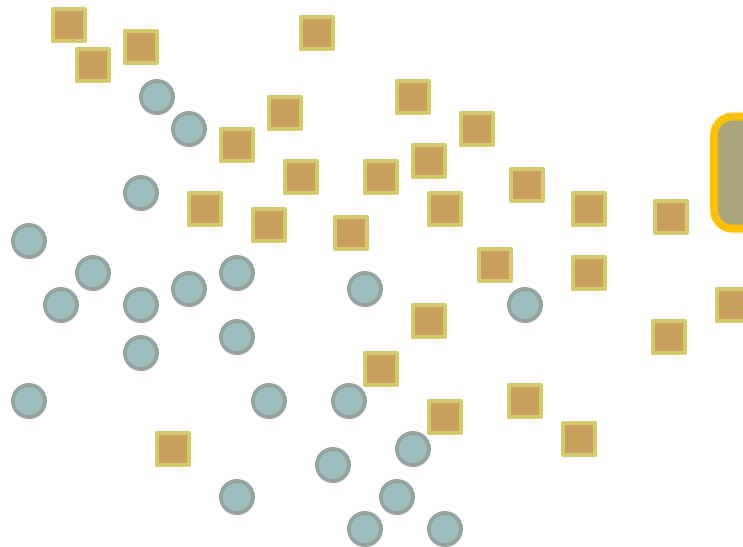
$$\text{isSquare} \sim \text{xLocation} + \text{yLocation}$$



Evaluate Model: Confusion Matrix



$\text{isSquare} \sim \text{xLocation} + \text{yLocation}$

Evaluate Model: Confusion Matrix



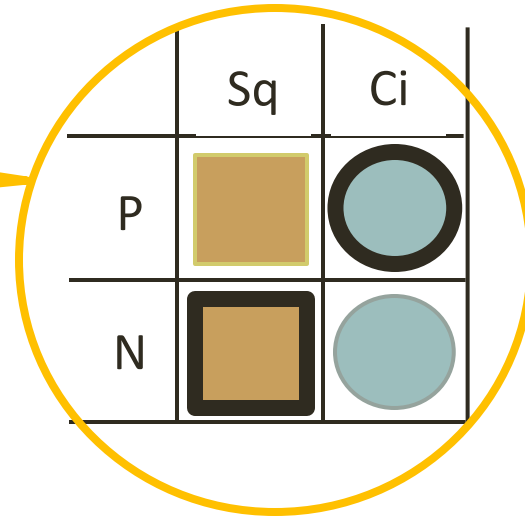
	Sq	Ci
P		
N		




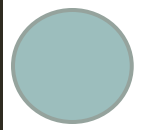
Incorrect Predictions are False Positives and False Negatives

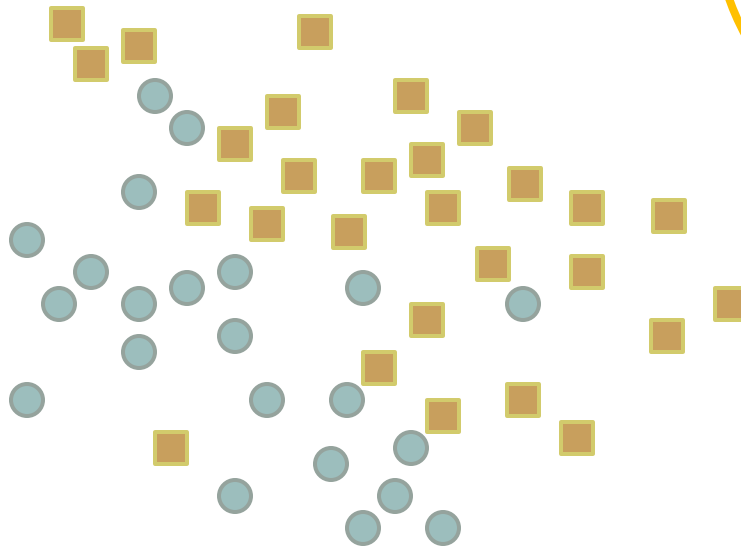
$$\text{isSquare} \sim \text{xLocation} + \text{yLocation}$$

Evaluate Model: Confusion Matrix

Confusion Matrix (Classification Matrix):
Vertical are actual classes
Horizontal are predicted classes

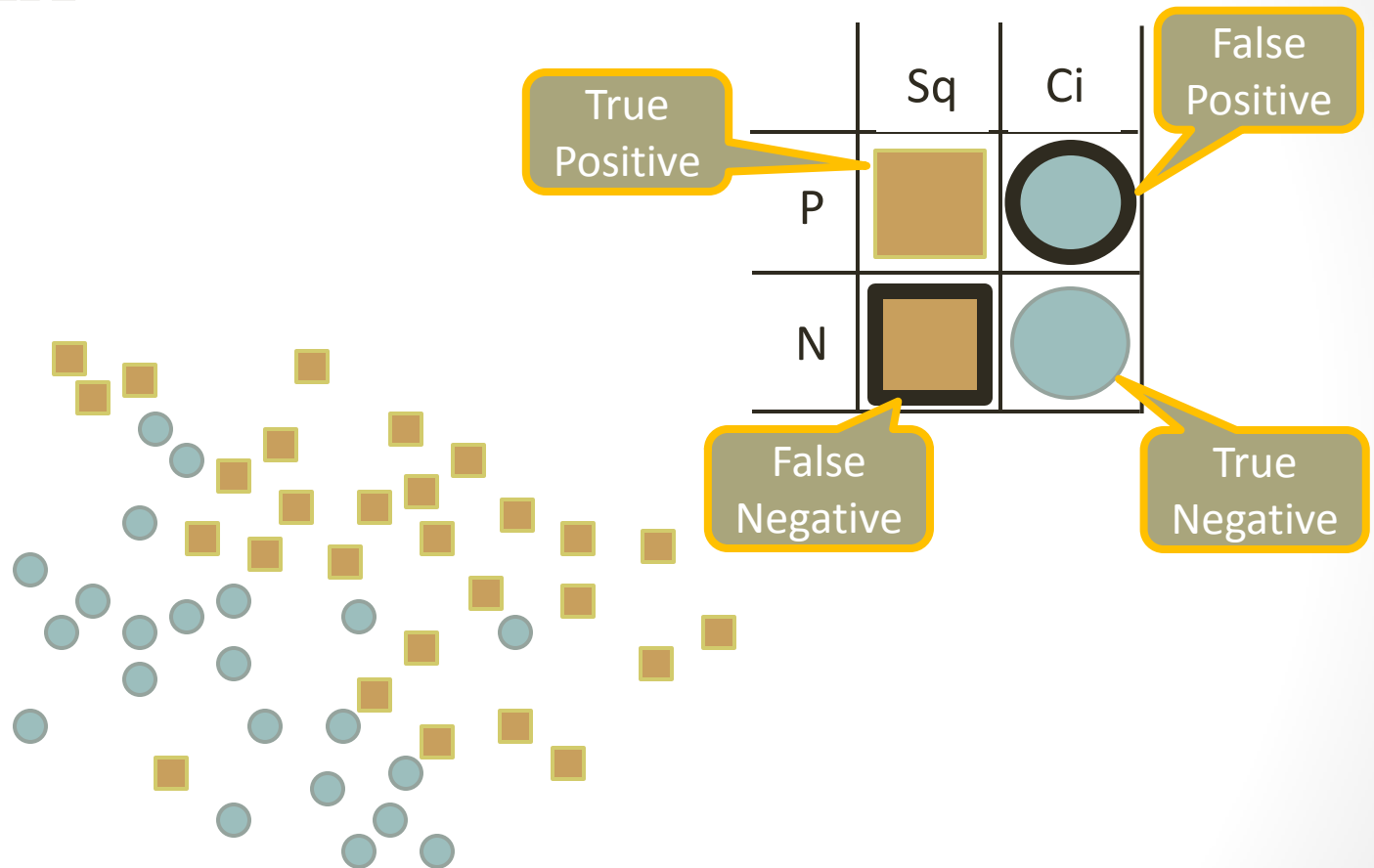


	Sq	Ci
P		
N		



$\text{isSquare} \sim \text{xLocation} + \text{yLocation}$

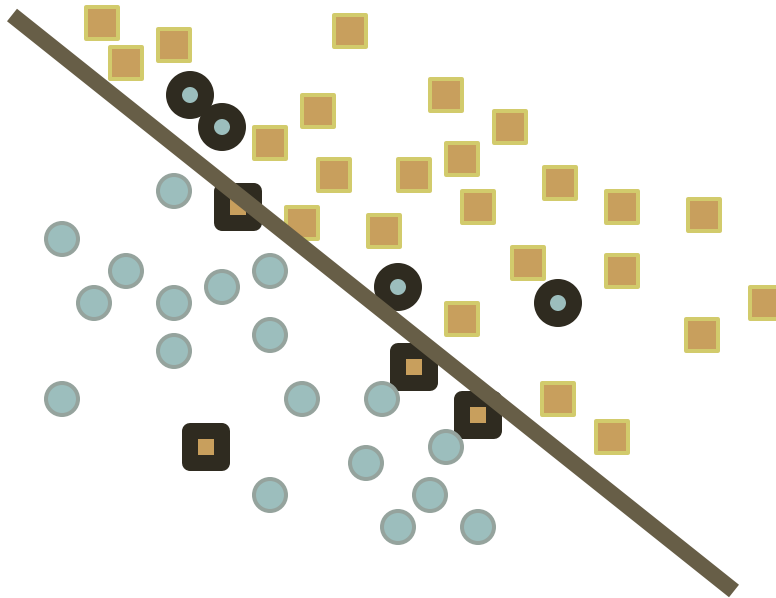
Evaluate Model: Confusion Matrix



$$\text{isSquare} \sim \text{xLocation} + \text{yLocation}$$

Evaluate Model : Train Model 1

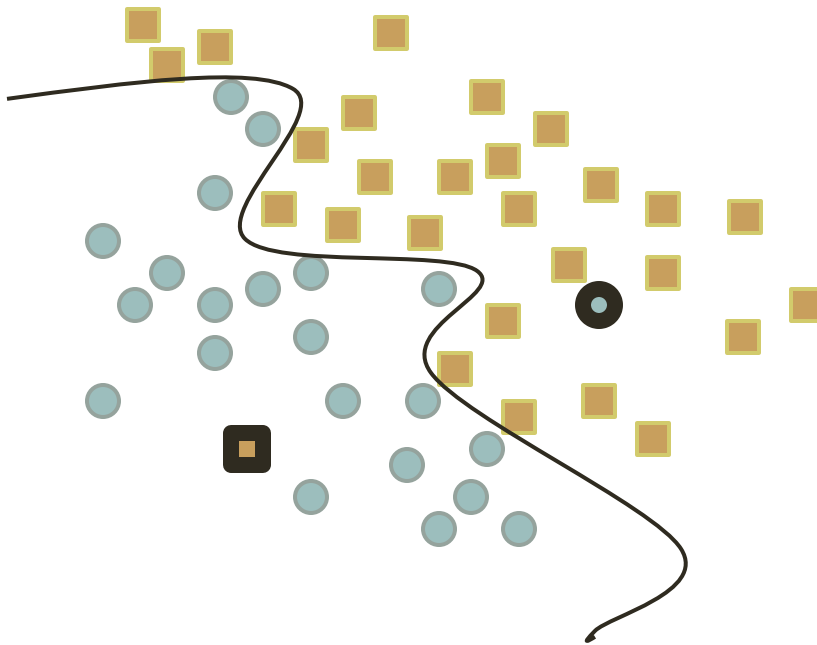
	Sq	Ci
P	36	4
N	4	26



$\text{isSquare} \sim x\text{Location} + y\text{Location}$

Evaluate Model : Train Model 2

	Sq	Ci
P	39	1
N	1	29



$\text{isSquare} \sim \text{xLocation} + \text{yLocation}$

Evaluate Model : Train Model 3

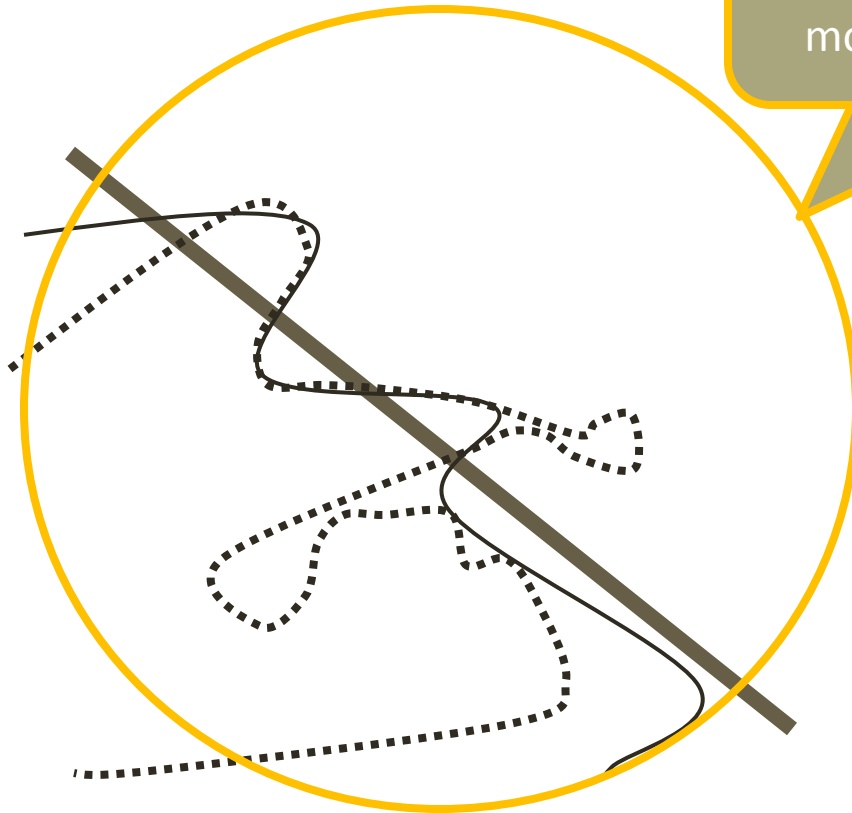
	Sq	Ci
P	40	0
N	0	30



$\text{isSquare} \sim x\text{Location} + y\text{Location}$

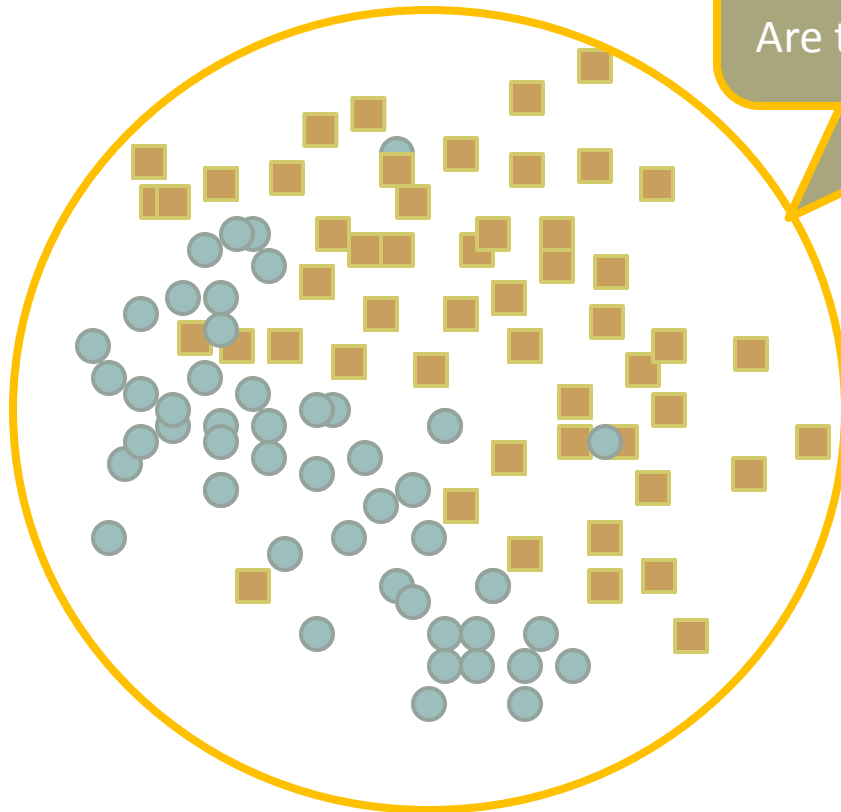
Evaluate Model : 3 Models

These models are based on training data. In these cases, models are called hypotheses.



$$\text{isSquare} \sim \text{xLocation} + \text{yLocation}$$

Evaluate Model : All Data



Training data overlaid on test data.
Visual comparison of data sets.
Are the distributions comparable?

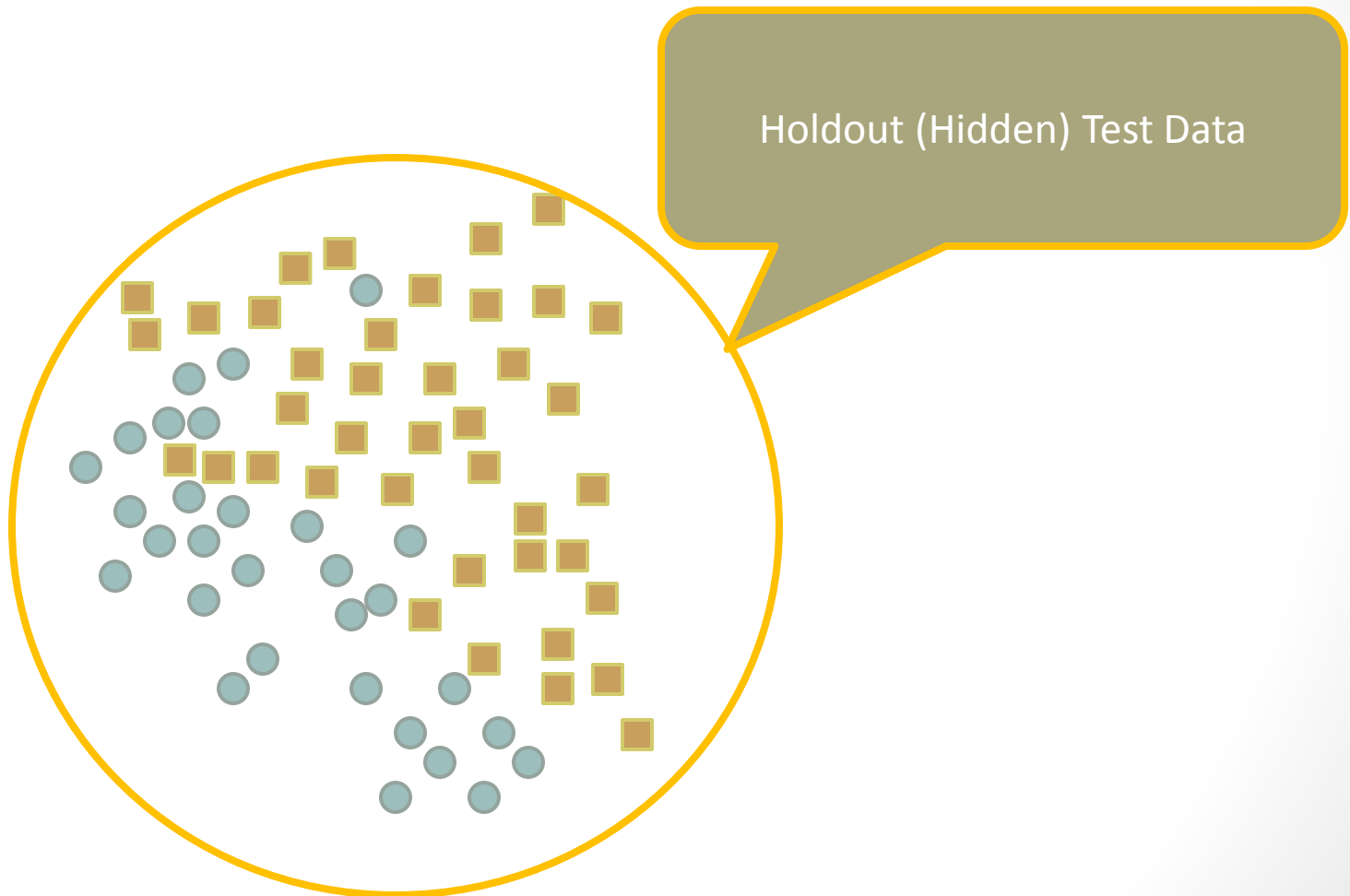
$$\text{isSquare} \sim \text{xLocation} + \text{yLocation}$$

Evaluate Model : Training Data



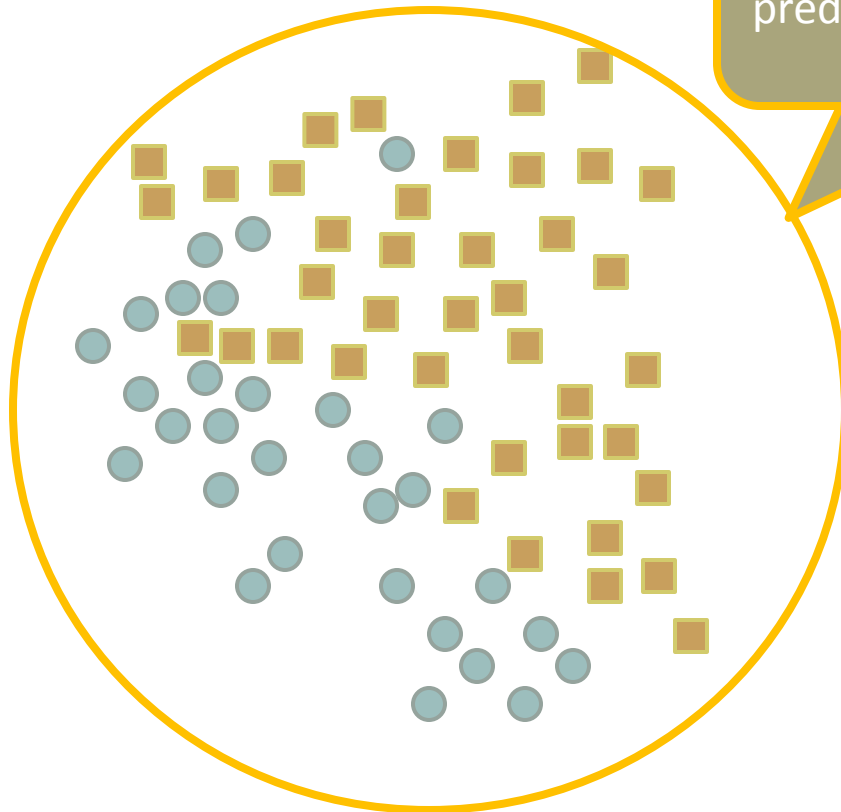
$$\text{isSquare} \sim \text{xLocation} + \text{yLocation}$$

Evaluate Model : Test Data



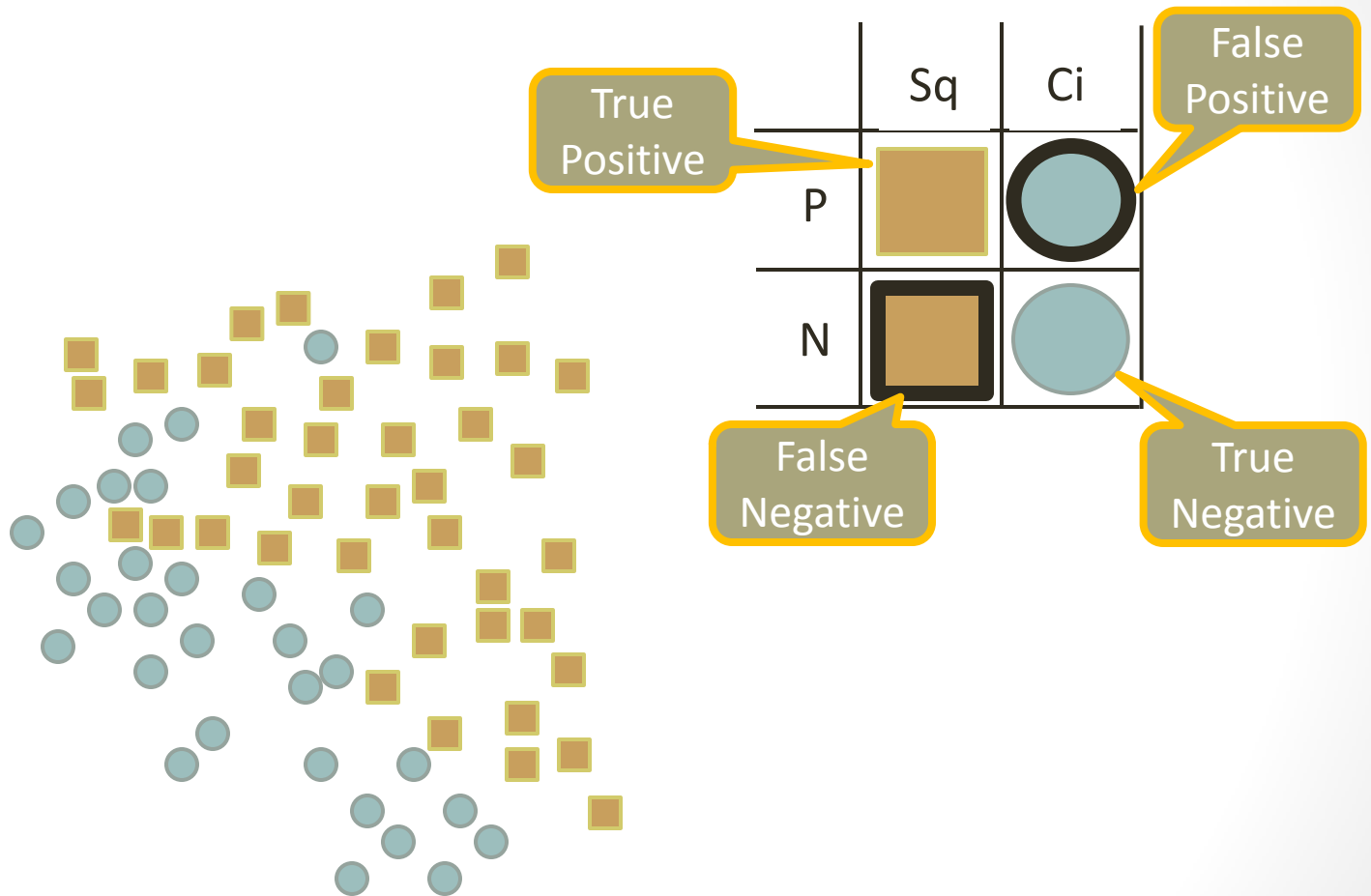
$$\text{isSquare} \sim x\text{Location} + y\text{Location}$$

Evaluate Model : Test Data



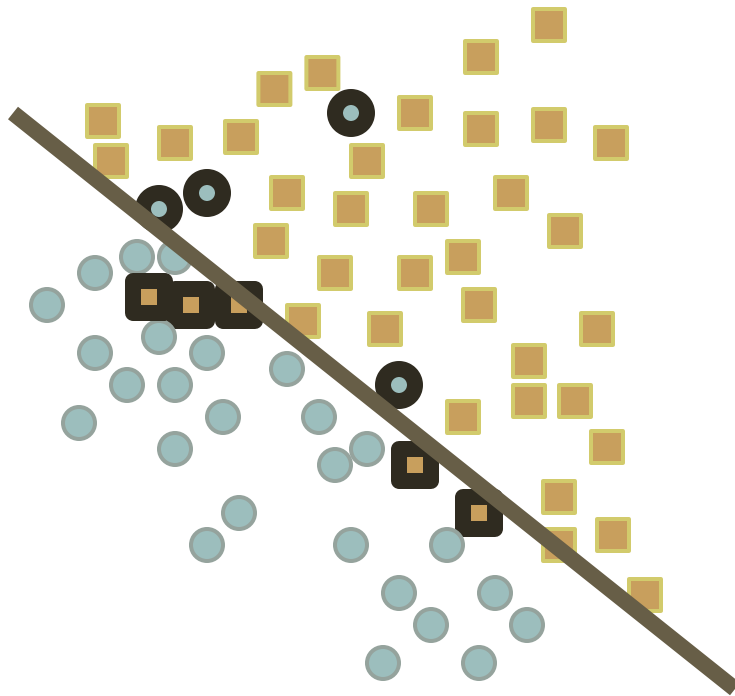
$\text{isSquare} \sim \text{xLocation} + \text{yLocation}$

Evaluate Model : Test Data



$$\text{isSquare} \sim x\text{Location} + y\text{Location}$$

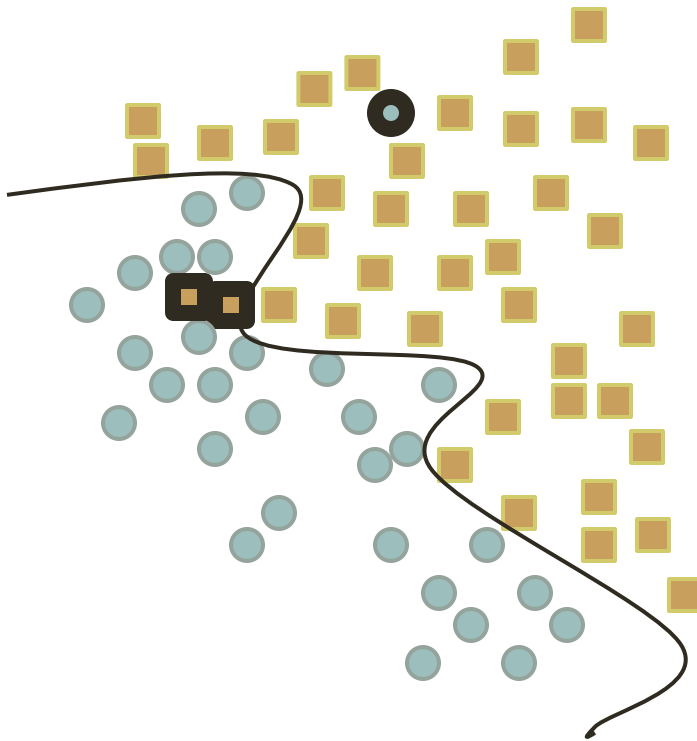
Evaluate Model : Test Model 1



	Sq	Ci
P	35	4
N	5	26

$\text{isSquare} \sim x\text{Location} + y\text{Location}$

Evaluate Model : Test Model 2

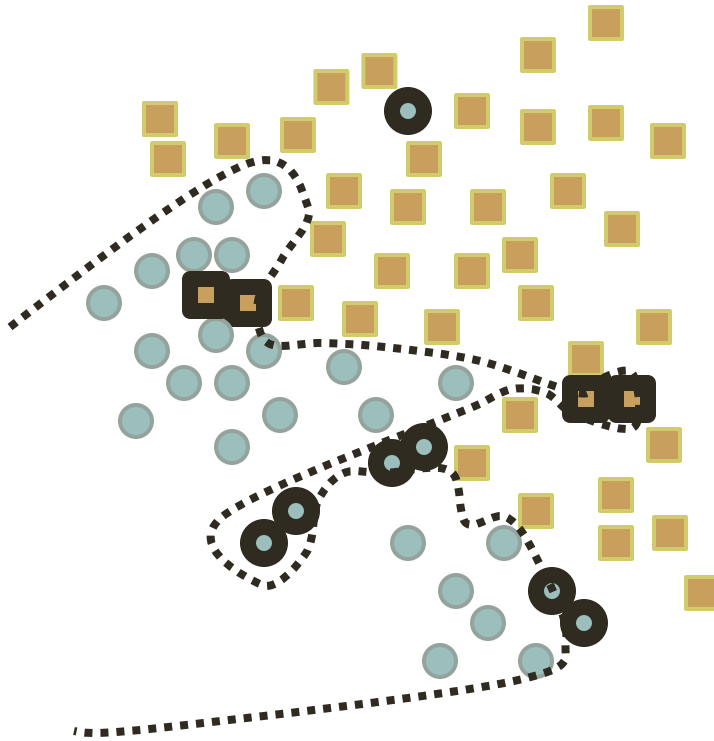


	Sq	Ci
P	38	1
N	2	29

$\text{isSquare} \sim x\text{Location} + y\text{Location}$

Evaluate Model : Test Model 3

	Sq	Ci
P	36	7
N	4	23



$\text{isSquare} \sim x\text{Location} + y\text{Location}$

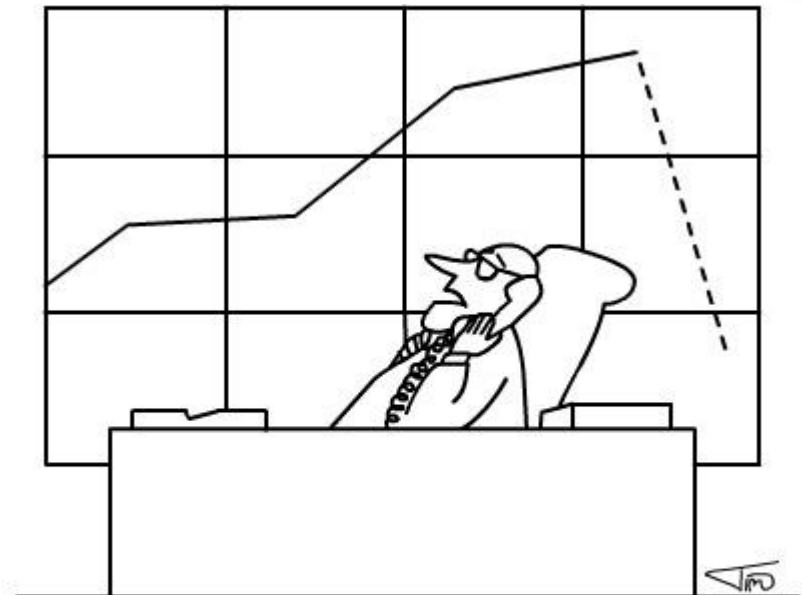
Relate a Confusion Matrix with an ROC chart

- Optional exercise in Predixion Insight: Open up a synced Classification (Confusion) Matrix and ROC chart.
 - Set the threshold of the Classification Matrix to 0, 0.5, and 1. How do these thresholds compare to the FPR and TPR on the ROC chart?
 - Set the FPR on the ROC chart to 0, 0.5, and 1. What are the TPR on the ROC chart? How does the threshold of the classification matrix change?
 - Open up a cost chart. Set the readmission penalty to 3X the cost of the intervention cost. What is the optimal threshold? What is the FPR?

Over-fitting and Confusion Matrix

Video and Break

- Watch in class this advertisement for IBM's predictive analytics: https://www.youtube.com/watch?v=v5m_g72KaKg
- Watch at home this very high-level introduction to predictive analytics: <https://www.youtube.com/watch?v=DJS-WvHmoB0>



"BI tech support? The predictive analysis system is giving the wrong answer again—can you please fix it?... "

How to make an ROC

How to make an ROC (0)

- From Probabilities to ROC:
- Probabilities -> Threshold -> Predictions -> Confusion Matrix -> ROC
- Get Excel workbook: [HowToMakeAnROC.xls](#)
- Note that at the bottom of the worksheet are the actual outcomes and the predicted probabilities.

Exercise: Threshold → Confusion Matrix → ROC (1)

Paste the actual outcomes and the predicted probabilities here.

	A	C	D	E	F	G	H	I	J	K
1		Predicted	Predicted							
2	Actual	Probability	Class	TP	FP	FN	TN	Threshold	FPR	TPR
3			0	0	0	0	1	0		
4			0	0	0	0	1	0.1		
5			0	0	0	0	1	0.2		
6			0	0	0	0	1	0.3		
7			0	0	0	0	1	0.4		
8			0	0	0	0	1	0.5		
9			0	0	0	0	1	0.6		
10			0	0	0	0	1	0.7		
11			0	0	0	0	1	0.8		
12			0	0	0	0	1	0.9		
13				0	0	0	10	1		
14										
15		TP	FP	0	0					
16		FN	TN	0	10					
17						Threshold:	0.5			
18						FPR:	0			
						TPR:	#DIV/0!			

Exercise: Threshold → Confusion Matrix → ROC (2)

Paste the actual outcomes and the predicted probabilities here

	A		C	D	E	F	G	H	I	J	K
1		Predicted	Predicted								
2	Actual	Probability	Class	TP	FP	FN	TN		Threshold	FPR	TPR
3	1	0.55	1	1	0	0	0		0	1	1
4	0	0.15	0	0	0	0	1		0.1		
5	1	0.65	1	1	0	0	0		0.2		
6	0	0.35	0	0	0	0	1		0.3		
7	1	0.15	0	0	0	1	0		0.4		
8	1	0.85	1	1	0	0	0		0.5		
9	0	0.25	0	0	0	0	1		0.6		
10	1	0.75	1	1	0	0	0		0.7		
11	0	0.55	1	0	1	0	0		0.8		
12	0	0.75	1	0	1	0	0		0.9		
13				4	2	1	3		1	0	0
14											
15	TP	FP		4	2						
16	FN	TN		1	3			Threshold:	0.5		
17								FPR:	0.4		
18								TPR:	0.8		

Exercise: Threshold → Confusion Matrix → ROC (3)

Microsoft Excel non-commercial use										
File	Home	Insert	Page Layout	Develop	Insight AI	Insight N	Team	?		
G16										
	A	B		G	H	I	J	K		
1		Predicted	Predicted							
2	Actual	Probability	Class	TP	FP	FN	TN	Threshold	FPR	TPR
3	1	0.55	1	1	0	0	0	0	1	1
4	0	0.15	0	0	0	0	1	0.1		
5	1	0.65	1	1	0	0	0	0.2		
6	0	0.35	0	0	0	0	1	0.3		
7	1	0.15	0	0	0	1	0	0.4		
8	1	0.85	1	1	0	0	0	0.5		
9	0	0.25	0	0	0	0	1	0.6		
10	1	0.75	1	1	0	0	0	0.7		
11	0	0.55	1	0	1	0	0	0.8		
12	0	0.75	1	0	1	0	0	0.9		
13				4	2	1	3	1	0	0
14										
15	TP	FP		4	2					
16	FN	TN		1	3					
17										
18										

The Predicted Probabilities need a threshold

Threshold: 0.5

FPR: 0.4

TPR: 0.8

Exercise: Threshold → Confusion Matrix → ROC (4)

Class_Confusion_ROC.xlsx - Microsoft Excel non-commercial use

File Home Insert Page Layout Formulas Data Review View Developer Insight AI Insight N Team

G16 fx 0.5

	A	B	C	D	E	F	G	H	I	J	K
1		Predicted	Predicted								
2	Actual	Probability	Class	TP	FP	FN	TN		Threshold	FPR	TPR
3	1	0.55	1	1	0	0	0		0	1	1
4	0	0.15	0	0	0	0	1		0.1		
5	1	0.65	1	1	0	0	0		0.2		
6	0	0.35	0	0	0	0	1		0.3		
7	1	0.15	0	0	0	1	0		0.4		
8	1	0.85	1	1	0	0	0		0.5		
9	0	0.25	0	0	0	0	1		0.6		
10	1	0.75	1	1	0	0	0		0.7		
11	0	0.55	1	0	1	0	0		0.8		
12	0	0.75	1	0	1	0	0		0.9		
13				4	2	1	3		1	0	0
14											
15	TP	FP		4	2						
16	FN	TN		1	3						
17											
18											

Set the threshold for the Predicted Probabilities

Threshold: 0.5

FPR: 0.4

TPR: 0.8

Sheet1 Sheet2 Sheet3

Ready 100%

Exercise: Threshold → Confusion Matrix → ROC (5)

Class = Probability > Threshold

	A	B	C	D	E	F	G	H	I	J	K
1		Predicted	Predicted								
2	Actual	Probability	Class	TP	FP	FN	TN	Threshold	FPR	TPR	
3	1	0.55	1	1	0	0	0	0	1	1	
4	0	0.15	0	0	0	0	1	0.1			
5	1	0.65	1	1	0	0	0	0.2			
6	0	0.35	0	0	0	0	1	0.3			
7	1	0.15	0	0	0	1	0	0.4			
8	1	0.85	1	1	0	0	0	0.5			
9	0	0.25	0	0	0	0	1	0.6			
10	1	0.75	1	1	0	0	0	0.7			
11	0	0.55	1	0	1	0	0	0.8			
12	0	0.75	1	0	1	0	0	0.9			
13				4	2	1	3	1	0	0	
14											
15	TP	FP		4	2						
16	FN	TN		1	3			Threshold	0.5		
17								FPR:	0.4		
18								TPR:	0.8		

Sheet1 Sheet2 Sheet3

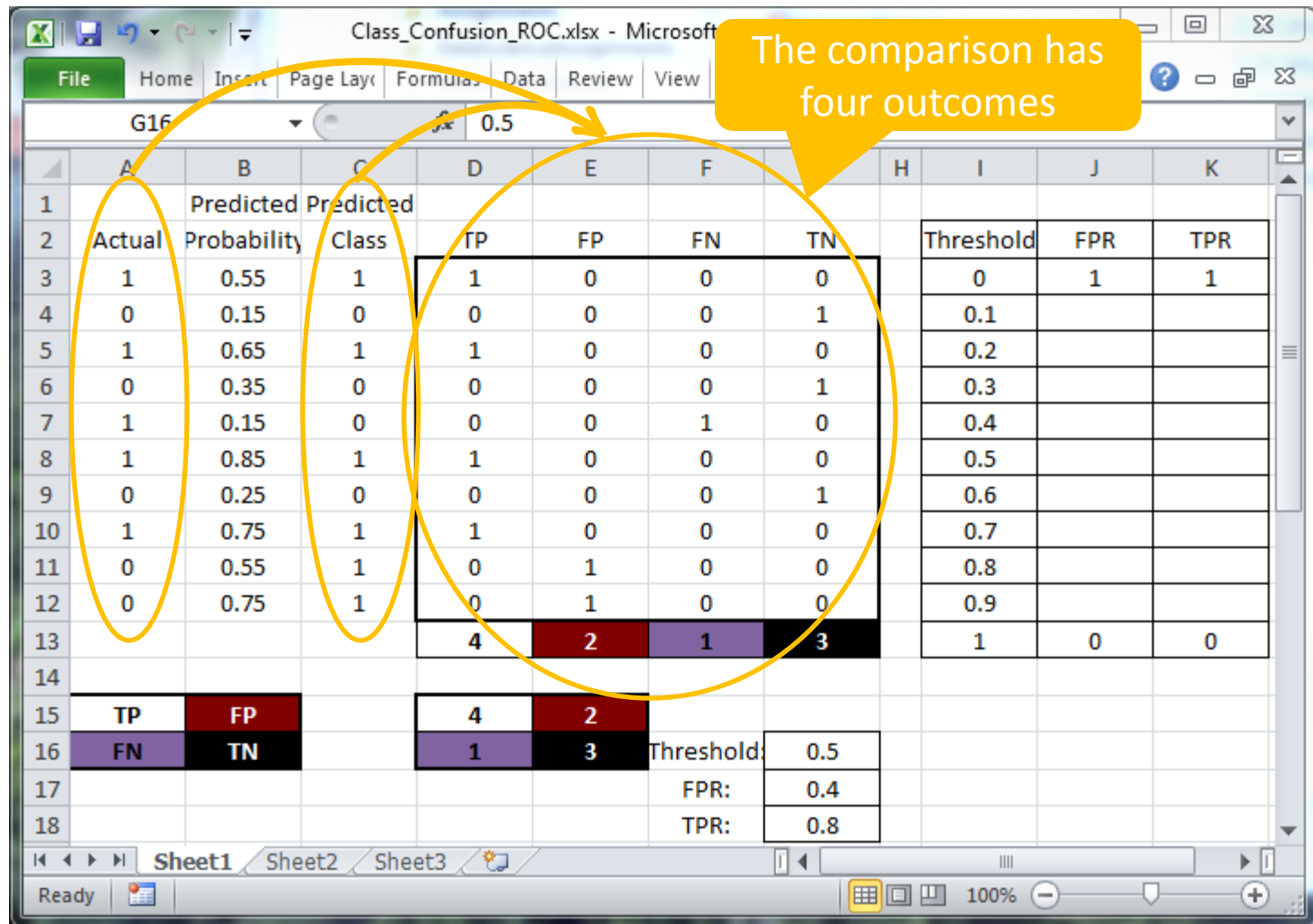
Exercise: Threshold → Confusion Matrix → ROC (6)

Compare the predicted Class to the Actual Values

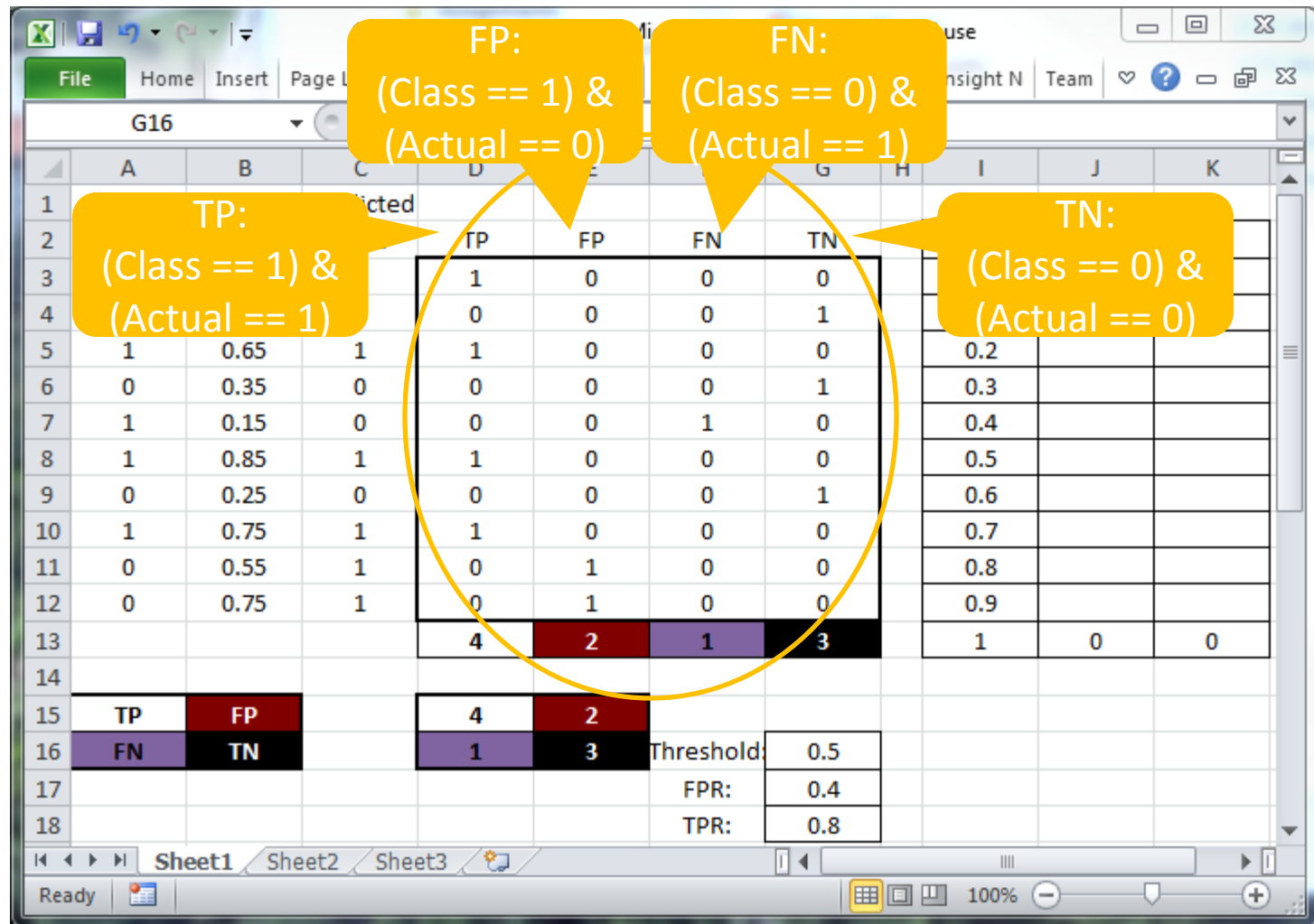
	A	B	C	D	E	F	G	H	I	J	K
1		Predicted	Predicted								
2	Actual	Probability	Class	TP	FP	FN	TN		Threshold	FPR	TPR
3	1	0.55	1	1	0	0	0		0	1	1
4	0	0.15	0	0	0	0	1		0.1		
5	1	0.65	1	1	0	0	0		0.2		
6	0	0.35	0	0	0	0	1		0.3		
7	1	0.15	0	0	0	1	0		0.4		
8	1	0.85	1	1	0	0	0		0.5		
9	0	0.25	0	0	0	0	1		0.6		
10	1	0.75	1	1	0	0	0		0.7		
11	0	0.55	1	0	1	0	0		0.8		
12	0	0.75	1	0	1	0	0		0.9		
13				4	2	1	3		1	0	0
14											
15	TP	FP		4	2						
16	FN	TN		1	3						
17											
18											

Threshold: 0.5
FPR: 0.4
TPR: 0.8

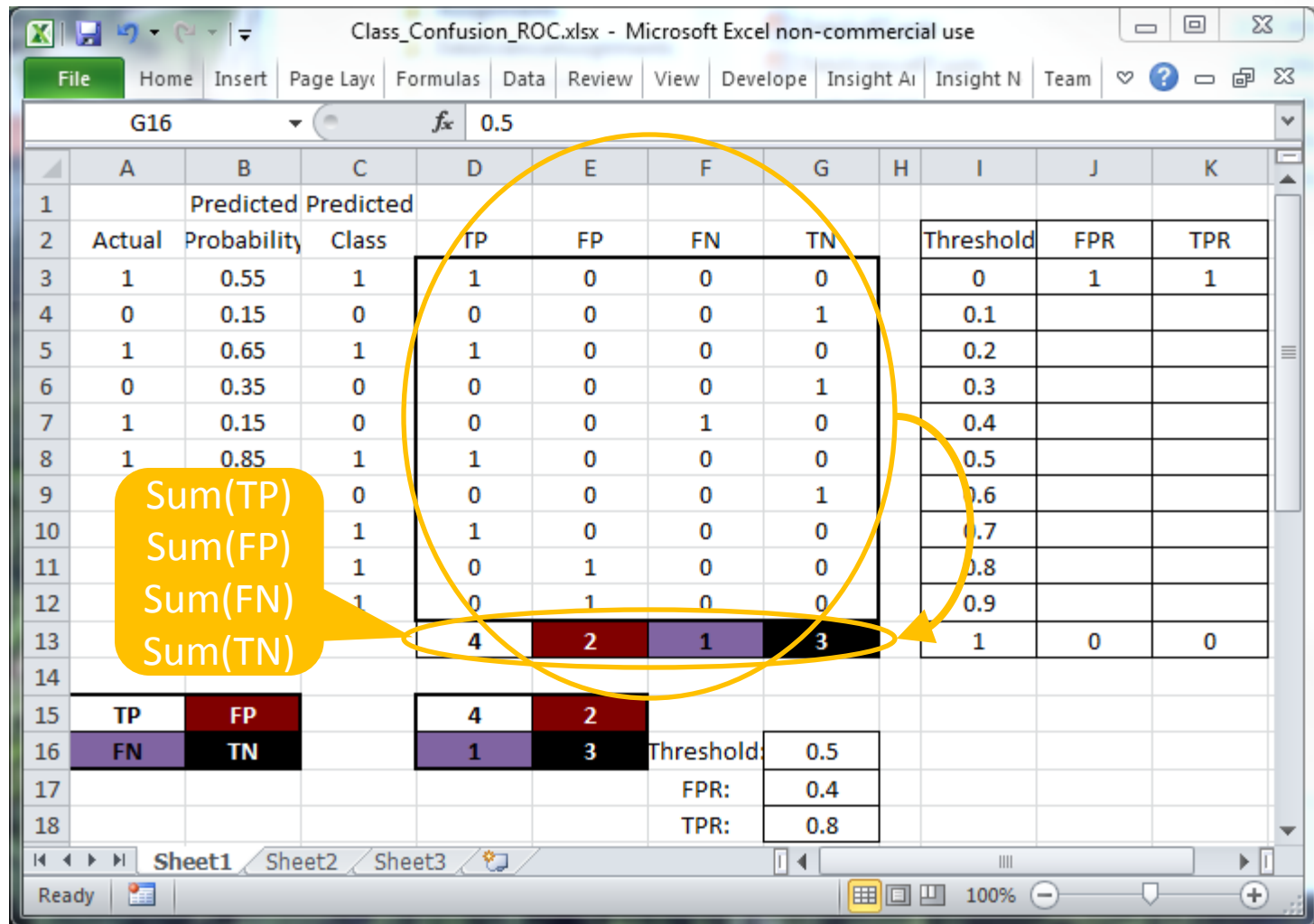
Exercise: Threshold → Confusion Matrix → ROC (7)



Exercise: Threshold → Confusion Matrix → ROC (8)

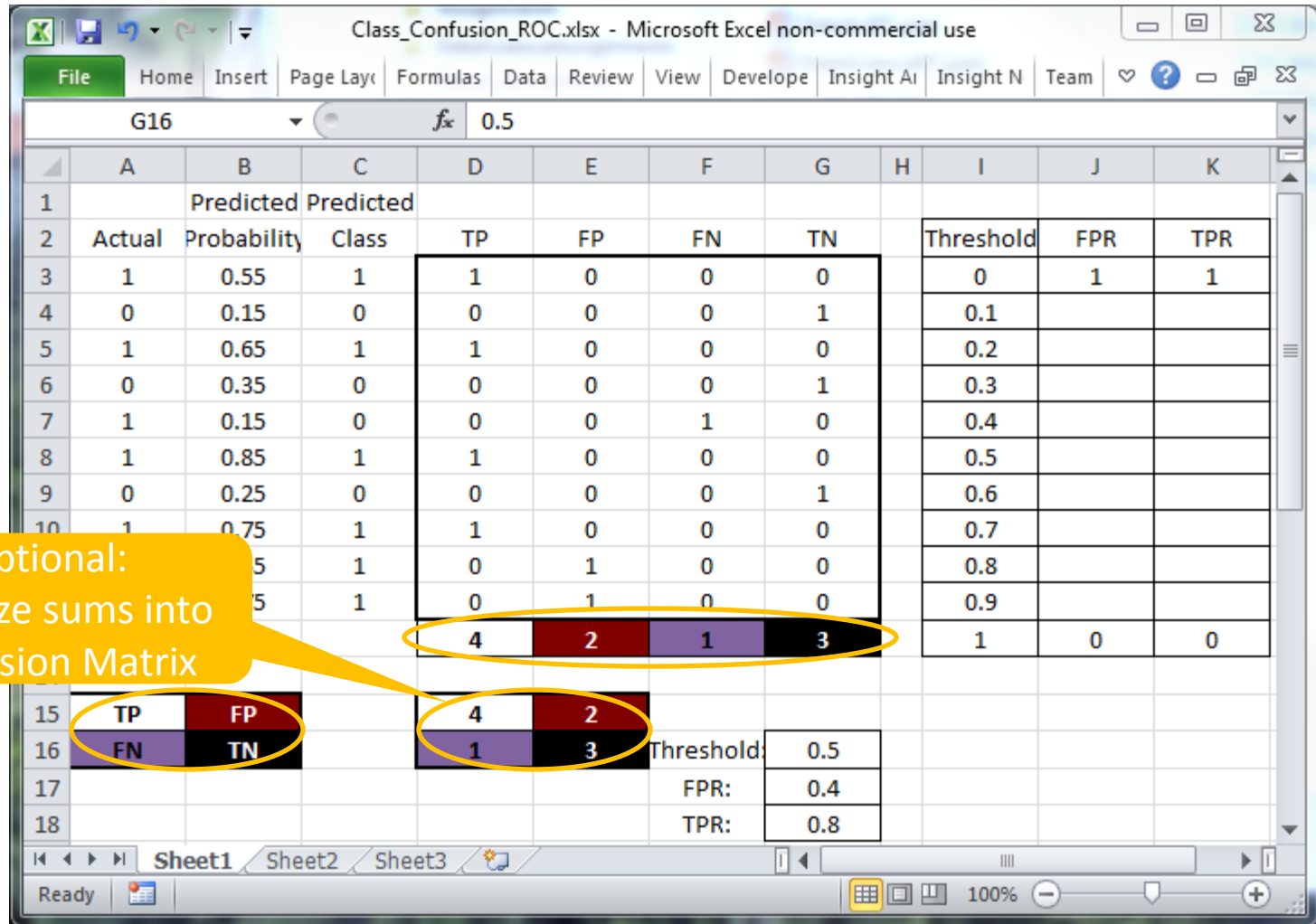


Exercise: Threshold → Confusion Matrix → ROC (9)



Exercise: Threshold \rightarrow

Confusion Matrix \rightarrow ROC (10)



Exercise: Threshold → Confusion Matrix → ROC (11)

Class_Confusion_ROC.xlsx - Microsoft Excel non-commercial use

File Home Insert Page Layout Formulas Data Review View Developer Insight AI Insight N Team

G16 fx 0.5

	A	B	C	D	E	F	G	H	I	J	K
1		Predicted	Predicted								
2	Actual	Probability	Class	TP	FP	FN	TN	Threshold	FPR	TPR	
3	1	0.55	1	1	0	0	0	0	1	1	
4	0	0.15	0	0	0	0	1	0.1			
5	1	0.65	1	1	0	0	0	0.2			
6	0	0.35	0	0	0	0	1	0.3			
7	1	0.15	0	0	0	1	0	0.4			
8	1	0.85	1	1	0	0	0	0.5			
9	0	0.25	0	0	0	0	1	0.6			
10	1	0.75	1	1	0	0	0	0.7			
11	0	0.55	1	0	1	0	0	0.8			
12	0	0.75	1	0	1	0	0	0.9			
13				4	2	1	3				
14											
15	TP	FP		4	2						
16	FN	TN		1	3						
17											
18											

TPR = TP / (TP + FN) = 4 / (4 + 1) = 0.8

FPR = FP / (FP + TN) = 2 / (2 + 3) = 0.4

Threshold: 0.5

FPR: 0.4

TPR: 0.8

Exercise: Threshold → Confusion Matrix → ROC (12)

Class_Confusion_ROC.xlsx - Microsoft Excel non-commercial use

File Home Insert Page Layout Formulas Data Review View Developer Insight AI Insight N Team

G16 fx 0.5

	A	B	C	D	E	F	G	H	I	J	K
1		Predicted	Predicted								
2	Actual	Probability	Class	TP	FP	FN	TN	Threshold	FPR	TPR	
3	1	0.55	1	1	0	0	0	0	1	1	
4	0	0.15	0	0	0	0	1	0.1			
5	1	0.65	1	1	0	0	0	0.2			
6	0	0.35	0	0	0	0	1	0.3			
7	1	0.15	0	0	0	1	0	0.4			
8	1	0.85	1	1	0	0	0	0.5			
9	0	0.25	0	0	0	0	1	0.6			
10	1	0.75	1	1	0	0	0	0.7			
11	0	0.55	1	0	1	0	0	0.8			
12	0	0.75	1	0	1	0	0	0.9			
13				4	2	1	3	1	0	0	
14											
15	TP	FP		4	2						
16	FN	TN		1	3			Threshold	0.5		
17								FPR:	0.4		
18								TPR:	0.8		

TPR = TP / (TP + FN)

Exercise: Threshold → Confusion Matrix → ROC (13)

Class_Confusion_ROC.xlsx - Microsoft Excel non-commercial use

File Home Insert Page Layout Formulas Data Review View Developer Insight AI Insight N Team

G16 fx 0.5

	A	B	C	D	E	F	G	H	I	J	K
1		Predicted	Predicted								
2	Actual	Probability	Class	TP	FP	FN	TN	Threshold	FPR	TPR	
3	1	0.55	1	1	0	0	0	0	1	1	
4	0	0.15	0	0	0	0	1	0.1			
5	1	0.65	1	1	0	0	0	0.2			
6	0	0.35	0	0	0	0	1	0.3			
7	1	0.15	0	0	0	1	0	0.4			
8	1	0.85	1	1	0	0	0	0.5			
9	0	0.25	0	0	0	0	1	0.6			
10	1	0.75	1	1	0	0	0	0.7			
11	0	0.55	1	0	1	0	0	0.8			
12	0	0.75	1	0	1	0	0	0.9			
13				4	2	1	3	1	0	0	
14											
15	TP	FP		4	2						
16	FN	TN		1	3			Threshold: 0.5			
17								FPR: 0.4			
18								TPR: 0.8			

Sheet1 Sheet2 Sheet3

Ready 100%

Exercise: Threshold → Confusion Matrix → ROC (14)

Class_Confusion_ROC.xlsx - Microsoft Excel non-commercial use

File Home Insert Page Layout Formulas Data Review View Developer Insight AI Insight N Team

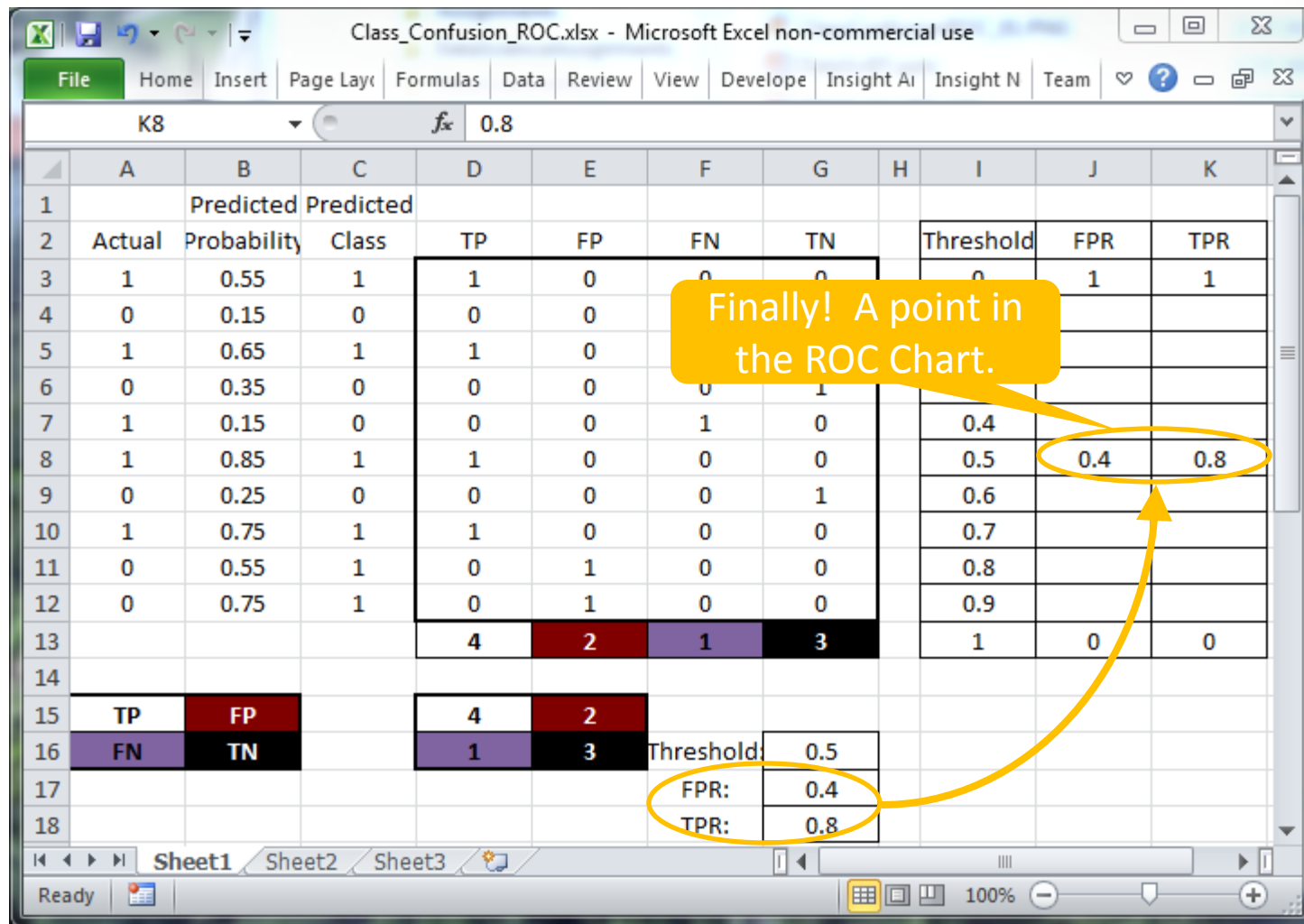
G16 fx 0.5

	A	B	C	D	E	F	G	H	I	J	K
1		Predicted	Predicted								
2	Actual	Probability	Class	TP	FP	FN	TN	Threshold	FPR	TPR	
3	1	0.55	1	1	0	0	0	0	1	1	
4	0	0.15	0	0	0	0	1	0.1			
5	1	0.65	1	1	0	0	0	0.2			
6	0	0.35	0	0	0	0	1	0.3			
7	1	0.15	0	0	0	1	0	0.4			
8	1	0.85	1	1	0	0	0	0.5			
9	0	0.25	0	0	0	0	1	0.6			
10	1	0.75	1	1	0	0	0	0.7			
11	0	0.55	1	0	1	0	0	0.8			
12	0	0.75	1	0	1	0	0	0.9			
13				4	2	1	3	1	0	0	
14											
15	TP	FP		4	2						
16	FN	TN		1	3	Threshold:	0.5				
17						FPR:	0.4				
18						TPR:	0.8				

Sheet1 Sheet2 Sheet3

Ready 100%

Exercise: Threshold → Confusion Matrix → ROC (15)



Exercise: Threshold → Confusion Matrix → ROC (16)

Class_Confusion_ROC.xlsx - Microsoft Excel non-commercial use

File Home Insert Page Layout Formulas Data Review View Developer Insight AI Insight N Team

K9 fx 0.6

	A	B	C	D	E	F	G	H	I	J	K
1		Predicted	Predicted								
2	Actual	Probability	Class	TP	FP	FN	TN		Threshold	FPR	TPR
3	1	0.55	0	0	0	1	0		0	1	1
4	0	0.15	0	0	0	0	1		0.1		
5	1	0.65	1	1	0	0	0		0.2		
6	0	0.35	0	0	0	0	1		0.3		
7	1	0.15	0	0	0	1	0		0.4		
8	1	0.85	1	1	0	0	0		0.5	0.4	0.8
9	0	0.25	0	0	0	0	1		0.6	0.2	0.6
10	1	0.75	1	1	0	0	0		0.7		
11	0	0.55	0	0	0	0	1		0.8		
12	0	0.75	1	0	1	0	0		0.9		
13				3	1	2	4		1	0	0
14											
15	TP	FP		3	1						
16	FN	TN		2	4						
17								Threshold:	0.6		
18								FPR:	0.2		
								TPR:	0.6		

Sheet1 Sheet2 Sheet3

Ready 100%

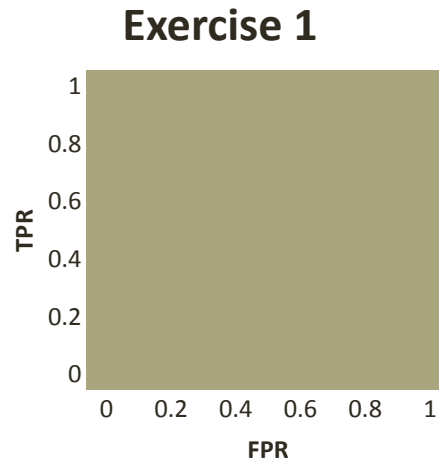
Repeat the process for all thresholds

Exercise: Threshold → Confusion Matrix → ROC (17)

Actual	Predicted Probability
1	0.55
0	0.15
1	0.65
0	0.35
1	0.15
1	0.85
0	0.25
1	0.75
0	0.55
0	0.75



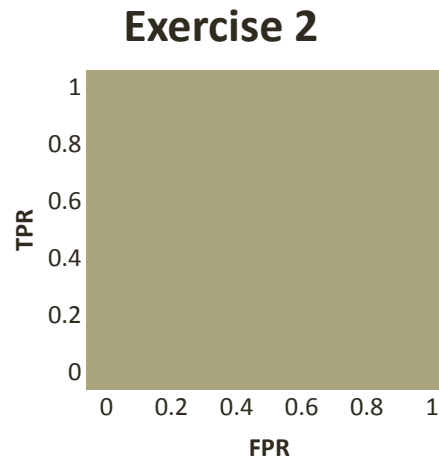
FPR	TPR
1	1
0	0



Actual	Predicted Probability
0	0.15
0	0.25
0	0.35
1	0.45
0	0.45
1	0.55
0	0.65
1	0.75
0	0.85
1	0.95



FPR	TPR
1	1
0	0



How to make an ROC

Quiz 06b

- <https://catalyst.uw.edu/webq/survey/ernsthe/270014>
- Test Measures Intro (ROC and Confusion Matrix)

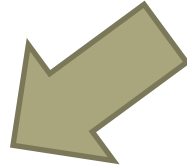


NoSQL

Scale-up vs. Scale-out

Scale Up vs. Scale Out

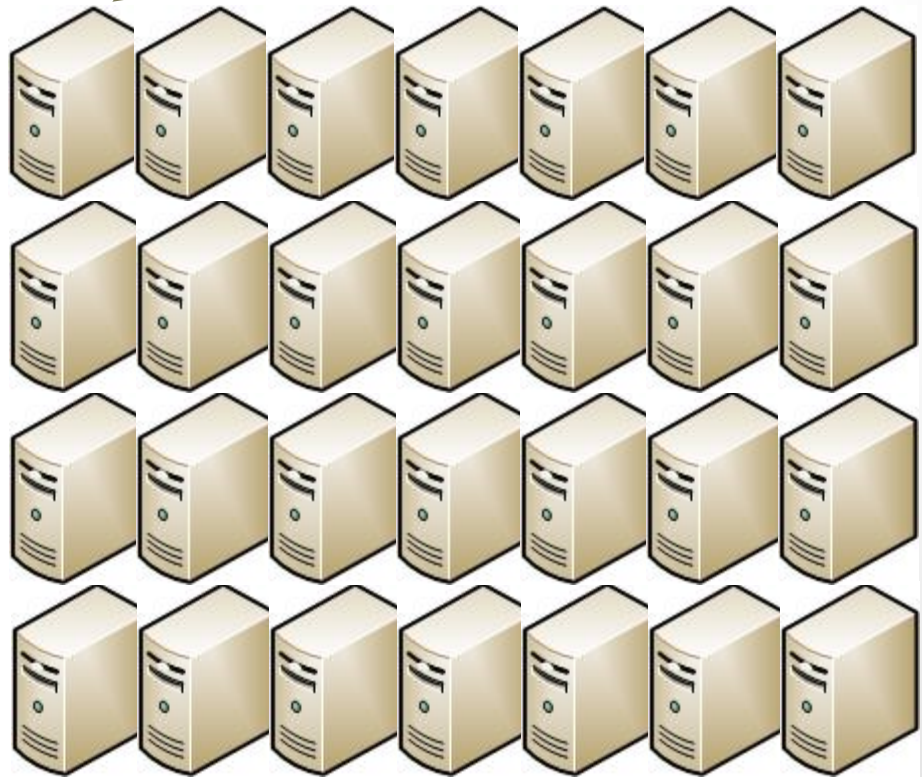
Scale Up



Scale Up vs. Scale Out

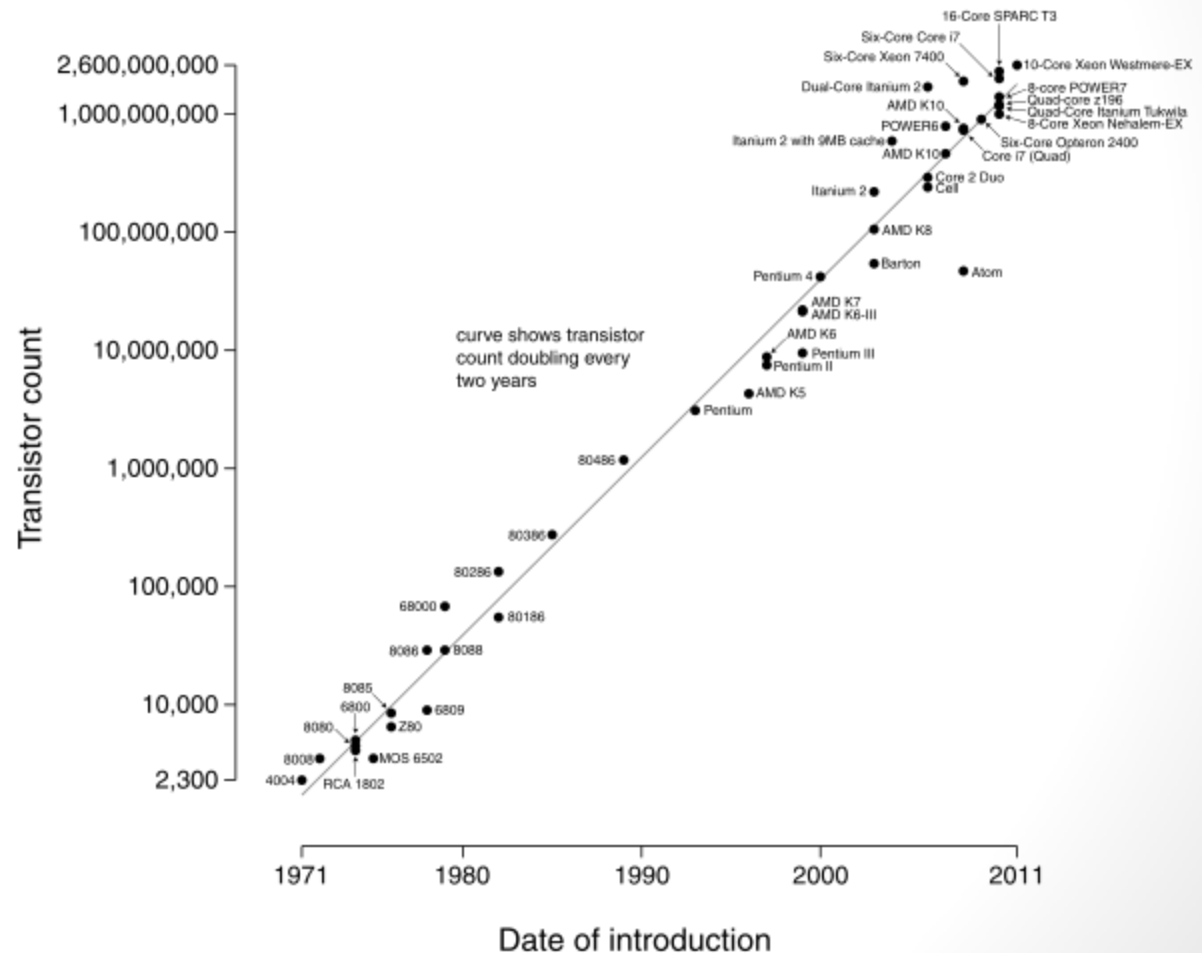
Scale Up

Scale Out



- Scale-up
- Moore's Law

Microprocessor Transistor Counts 1971-2011 & Moore's Law



Scale-up vs. Scale-out



Grace Hopper

Scale-up vs. Scale-out



Grace Hopper

"In pioneer days they used oxen for heavy pulling, and when one ox couldn't budge a log, they didn't try to grow a larger ox. We shouldn't be trying for bigger computers, but for more systems of computers."

Cloud: Scale-out

- The primary characteristic of NoSQL is scale out.
- From a practical level, scale out requires an adjustable number of commodity computers.
- Cluster Elasticity:
[http://en.wikipedia.org/wiki/Elasticity %28data store%29](http://en.wikipedia.org/wiki/Elasticity_%28data_store%29)
- Virtual Machine
 - One computer “mimics” another computer. (A system platform supports execution of an operating system)
 - Allows hardware standardization.
 - Allows one server to “host” many computers.
 - Virtual machines in the cloud can be set up and taken down (dehydrated, reduced to an image).
- Cloud: What is the “cloud”? Remote access to a single point provides many online services like servers and storage.
([http://en.wikipedia.org/wiki/Cloud computing](http://en.wikipedia.org/wiki/Cloud_computing)).

Cloud: Services

- Amazon Web Services
- GoGrid
- Google Compute Engine
- Microsoft Azure
- Rackspace
- SoftLayer



Scale-out and the “Cloud”

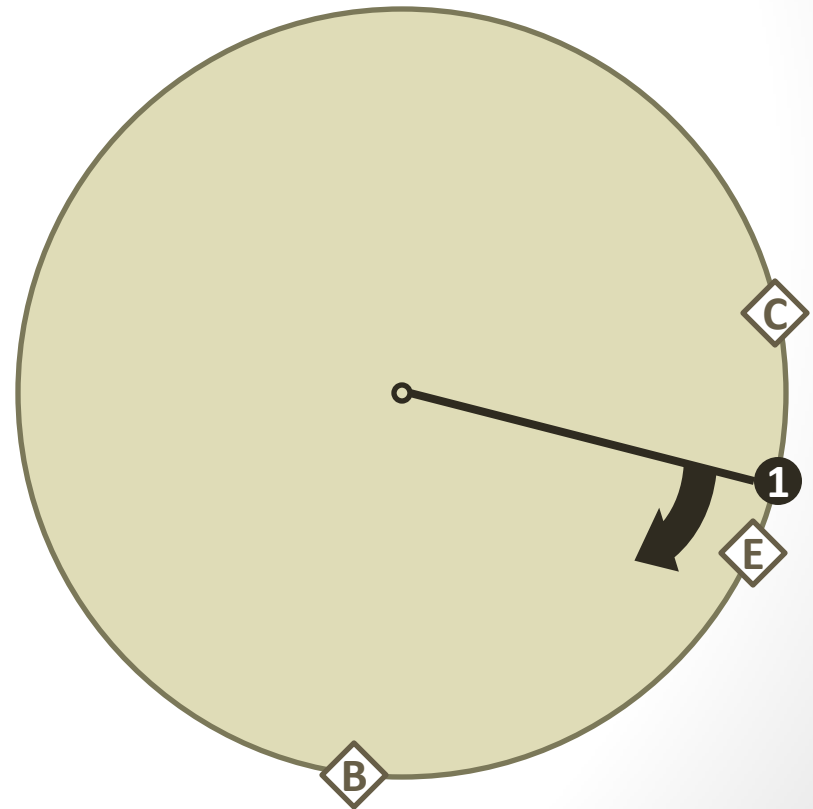
- **Elasticity** has made cloud computing feasible
- Clouds generally employ **virtual machines** that can be created at a moments notice, reduced to an image (dehydrated), re-started from an image, and deleted (recycled).
- How do we partition storage or usage among an unknown number of machines? Often we do not know ahead of time if new machines will become available or which machines will be recycled.
- Storage and usage are mapped to machines by a hash table. In traditional hash tables a change in the number of slots requires most keys to be remapped.
- We need a strategy to minimize remapping of storage and usage among the available computers: Consistent Hashing:
[http://en.wikipedia.org/wiki/Consistent hashing](http://en.wikipedia.org/wiki/Consistent_hashing)

Consistent Hashing

- Consider a hash map where each object is mapped to a point on the circumference of a circle. For instance an object is mapped to the number of minutes on a clock.
- Computers, Files, Processes, etc., are mapped in this manner on the same circle.
- A computer “claims” all files and processes who have a hash that is clock wise to that computer.

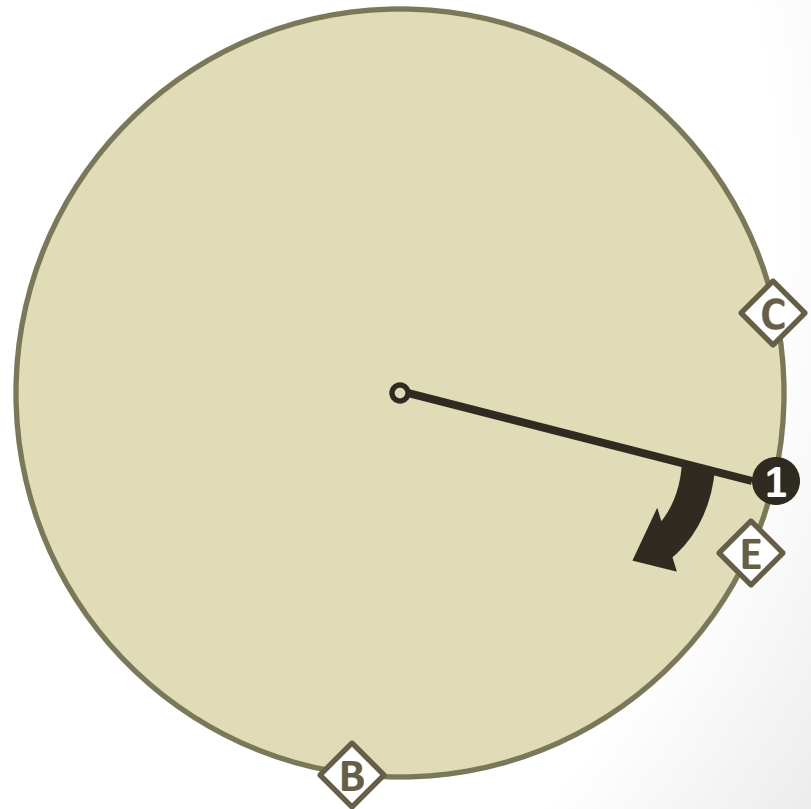
Consistent Hashing

Symbol	Object Type	Hash	Relation
B	Data Object	32	1
C	Data Object	14	1
E	Data Object	18	1
1	Machine 1	17	E C B



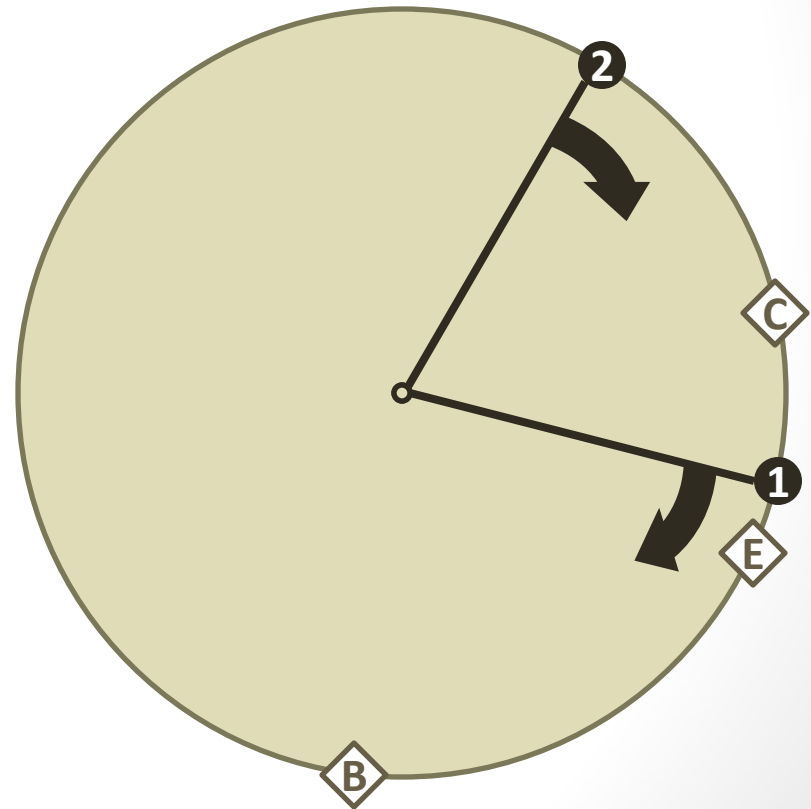
Consistent Hashing

Symbol	Object Type	Hash	Relation
B	Data Object	32	1
C	Data Object	14	1
E	Data Object	18	1
1	Machine 1	17	E C B



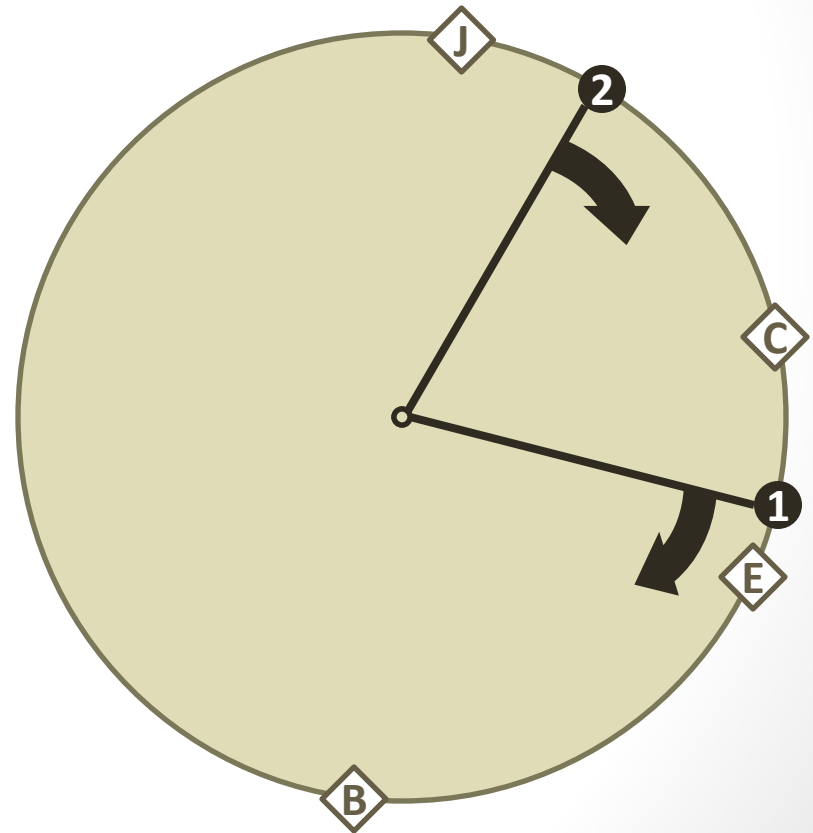
Consistent Hashing

Symbol	Object Type	Hash	Relation
B	Data Object	32	1
C	Data Object	14	2
E	Data Object	18	1
1	Machine 1	17	E B
2	Machine 2	5	C



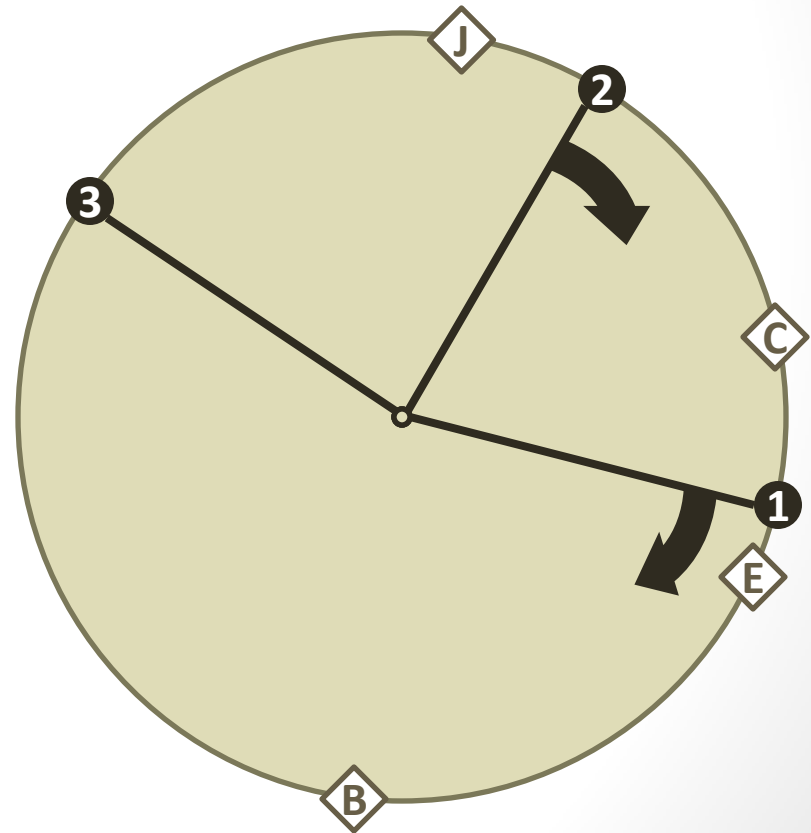
Consistent Hashing

Symbol	Object Type	Hash	Relation
B	Data Object	32	1
C	Data Object	14	2
E	Data Object	18	1
J	Data Object	2	1
1	Machine 1	17	E J B
2	Machine 2	5	C



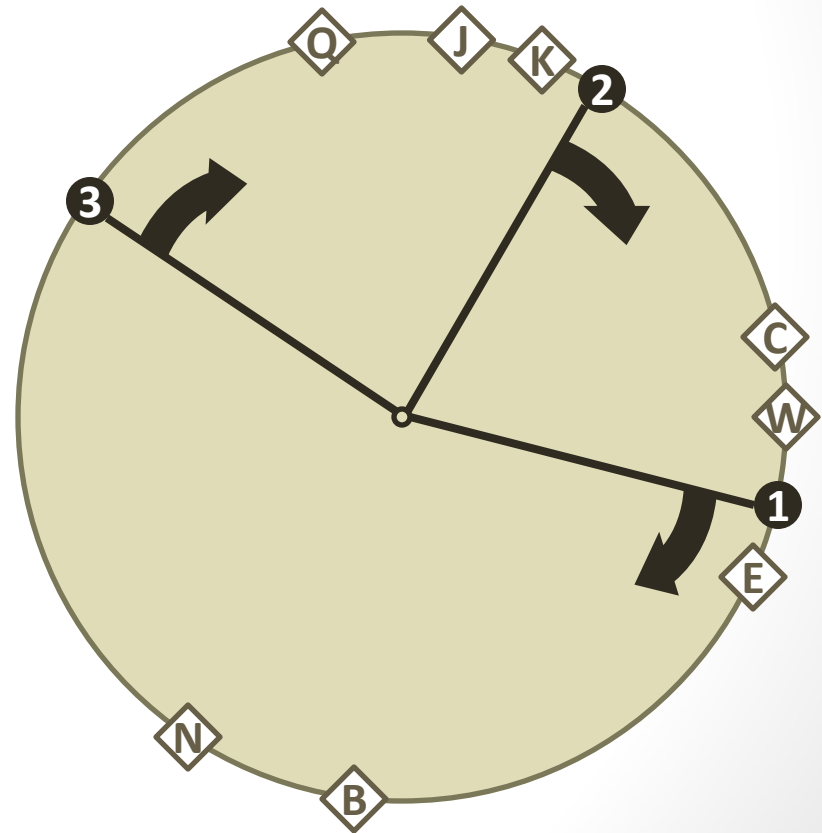
Consistent Hashing

Symbol	Object Type	Hash	Relation
B	Data Object	32	1
C	Data Object	14	2
E	Data Object	18	1
J	Data Object	2	3
1	Machine 1	17	E B
2	Machine 2	5	C
3	Machine 3	51	J



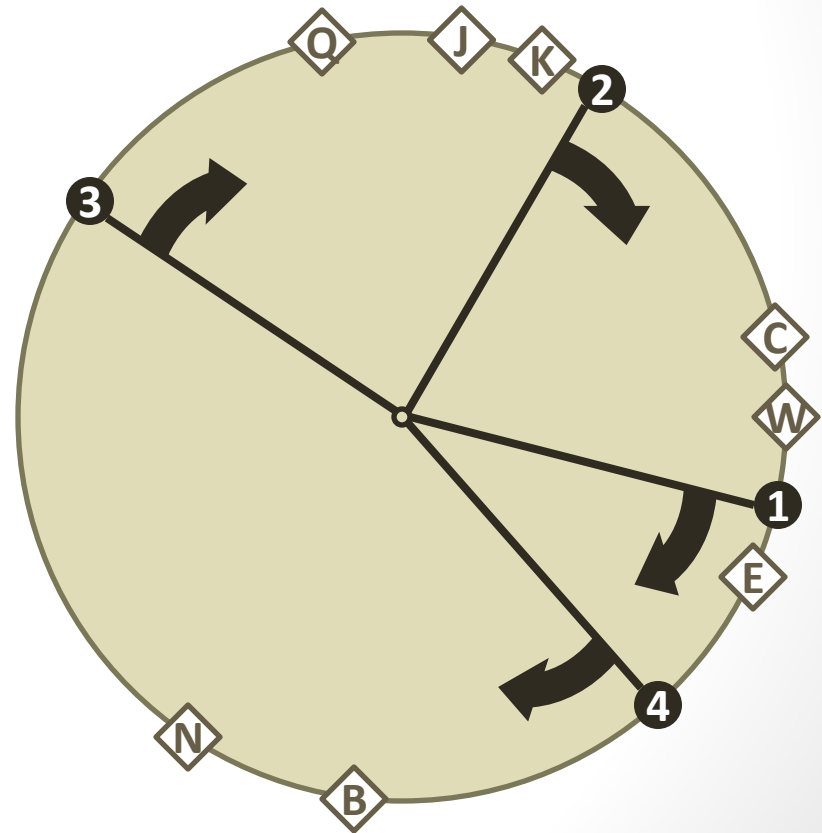
Consistent Hashing

Symbol	Object Type	Hash	Relation
B	Data Object	32	1
C	Data Object	14	2
E	Data Object	18	1
J	Data Object	2	3
K	Data Object	4	3
N	Data Object	35	1
Q	Data Object	57	3
W	Data Object	15	2
1	Machine 1	17	E B N
2	Machine 2	5	C W
3	Machine 3	51	J K Q



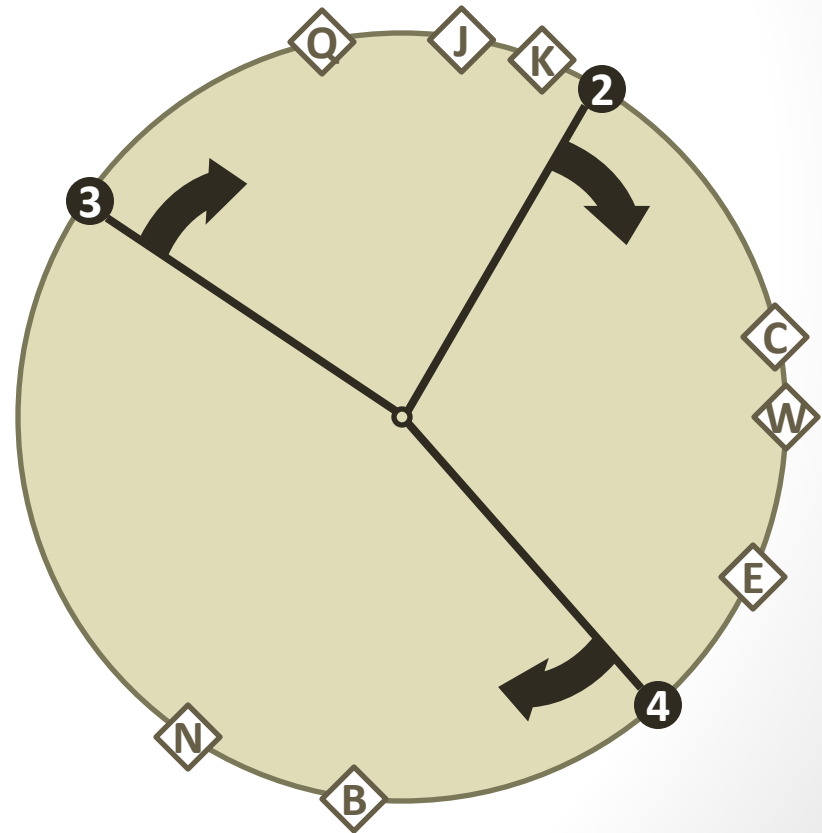
Consistent Hashing

Symbol	Object Type	Hash	Relation
B	Data Object	32	4
C	Data Object	14	2
E	Data Object	18	1
J	Data Object	2	3
K	Data Object	4	3
N	Data Object	35	4
Q	Data Object	57	3
W	Data Object	15	2
1	Machine 1	17	E
2	Machine 2	5	C W
3	Machine 3	51	J K Q
4	Machine 4	23	B N



Consistent Hashing

Symbol	Object Type	Hash	Relation
B	Data Object	32	4
C	Data Object	14	2
E	Data Object	18	2
J	Data Object	2	3
K	Data Object	4	3
N	Data Object	35	4
Q	Data Object	57	3
W	Data Object	15	2
2	Machine 2	5	C W E
3	Machine 3	51	J K Q
4	Machine 4	23	B N



What does Scale-Out have to do with NoSQL?

- Traditional Relational Database Management Systems (RDBMS) have problems with scale-out.
- Therefore, new data base management schemes were desired.

NoSQL

- NoSQL
 - **NO**SQL may stand for: **NO**T-SQL, **Not-Only**-SQL, **KNOW**-SQL
 - There is no consensus definition of NoSQL. NoSQL is a misnomer. No SQL has less to do with SQL or an alternative to query language. NoSQL has more to do with new database strategies and data structures.

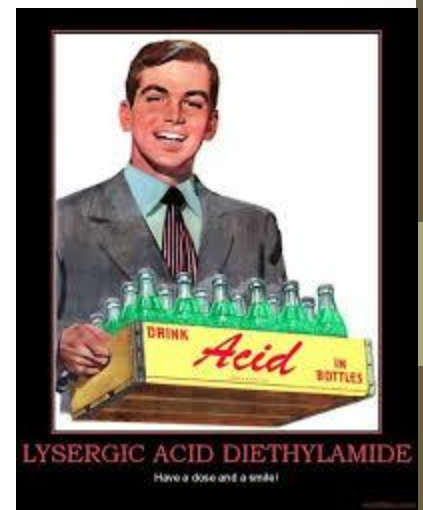


NoSQL

- NoSQL
 - **NO**SQL may stand for: **NO**T-SQL, **N**ot-**O**nly-SQL, **KNO**W-SQL
 - There is no consensus definition of NoSQL. NoSQL is a misnomer. NoSQL has less to do with SQL or an alternative to query language. NoSQL has more to do with new database strategies and data structures.
- RDBMS vs. NoSQL
 - NoSQL has to do with databases that do not follow the pattern of a relational database management system (RDBMS)
 - Therefore we need to define NoSQL in contrast to RDBMS. The hallmark of RDBMS is the relational model and **ACID**.
- Quick and Simple Overview of NoSQL (watch at home):
http://www.youtube.com/watch?v=sh1YACOK_bo

ACID: RDBMS

- ACID and the relational model are the hallmarks of RDBMS. ACID stands for:
 - Atomic
 - Consistent
 - Isolation
 - Durability



ACID: Atomic

- Atomic is Greek for unsplittable
- All or nothing
- All the changes of a transaction will happen or none of them will happen.
- Aborted transactions are rolled back.

ACID: Consistent

- Database is consistent before transaction. Database is consistent after transaction.
- Database will adhere to all the consistency rules before and after every transaction.
- Database constraints and column relations to other data are maintained. In other words, data written to the database must abide by integrity constraints. For Example:
 - A column which requires a unique identifier will not tolerate a duplicate value.
 - A column that requires no NULL values will not accept a NULL value.
 - The database will verify that each value is a valid foreign key in a column that demands that each value is a valid foreign key.

Beware! The word consistency is overloaded. The C in ACID is for a different consistency than the well-known consistency problem in a NoSQL distributed system. When we talk about consistency in NoSQL we mean that replicated data are the same after updates.

ACID: Isolation

- Transactions are isolated from one another.
- During a transaction, other processes cannot see the affected parts of the database until the transaction has completed. The other processes have to wait. The result is as if the transactions occurred in sequentially
- Isolation is achieved by concurrency control. When two transactions execute at the same time, each attempting to modify the same data, one of the two must wait until the other completes.

ACID: Durability

- What is written is readable until explicitly deleted
- Data doesn't evaporate

Durability is the hallmark of databases in general. NoSQL and RDBMS both succeed equally well in durability.

RDBMS vs. NoSQL

- The atomic, consistent, and isolated aspects of an RDBMS are the basis of what is called a transaction shell or bubble.
- Durability is just as important in NoSQL as it is in an RDBMS
- Base
 - Basic Availability: Basic Availability means that the system is available most of the time. (Availability means that a database request receives a response about success or failure.)
 - Soft-state
 - Eventual consistency

Beware! The word consistency is overloaded. The C in ACID is for a different consistency than the well-known consistency problem in a NoSQL distributed system. When we talk about consistency in NoSQL we mean that replicated data are the same after updates.

NoSQL

- NoSQL databases are distributed databases that split up data into manageable blocks and replicate data to prevent data loss
- NoSQL databases allow scale-out using many cheap servers and, typically, do not fully use scaled-up servers
- NoSQL databases may have a relaxed schema and can dynamically add new attributes to records
- NoSQL databases have a relaxed transaction shell and do not abide by ACID
- NoSQL databases do not need to be immediately consistent after every transaction. They can be eventually Consistent.

Beware! The word consistency is overloaded. The C in ACID is for a different consistency than the well-known consistency problem in a NoSQL distributed system. When we talk about consistency in NoSQL we mean that replicated data are the same after updates.

Break

CAP Theorem

- Continue at 8:43 PM



CAP Theorem

Distributed system with Shared Data: Vasanti Bhat-Nayak and Grace Hopper need a package from R to do a naïve Bayes classification. If there were only one server that contained this package, then consistency would be easy. But, availability would be restricted. When multiple R users want to download a package, the server gets clogged. Therefore, the cran packages are replicated on multiple servers around the world. When a package needs to be updated, then the master node asks all servers to update simultaneously. So when Vasanti and Grace download a package from different servers they will get the same version of the Naive Bayes package.

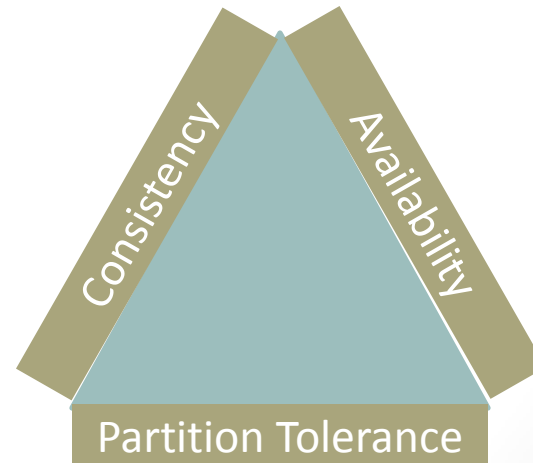
CAP Theorem

Distributed system with Shared Data: Vasanti Bhat-Nayak and Grace Hopper need a package from R to do a naïve Bayes classification. If there were only one server that contained this package, then consistency would be easy. But, availability would be restricted. When multiple R users want to download a package, the server gets clogged. Therefore, the cran packages are replicated on multiple servers around the world. When a package needs to be updated, then the master node asks all servers to update simultaneously. So when Vasanti and Grace download a package from different servers they will get the same version of the Naive Bayes package.

Partition of the Distributed System: But, what happens if on that day the Andorran server that Vasanti uses, can't be updated because of a communication error. The database has two choices: (1) It can wait until the Andorran server is fixed and then do the update. (2) Or, it updates all the other servers that allow the update. In the first case we forgo availability and nobody has access to the most recent Naive Bayes package. In the second case Vasanti and Grace will have different results because the packages are different.

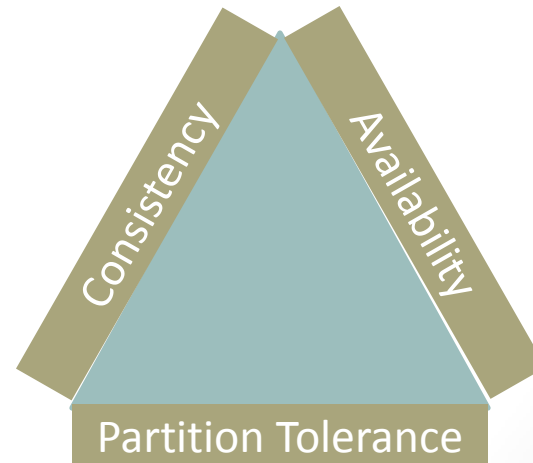
CAP Theorem

- CAP stands for:
 - Consistency
 - Availability
 - Partition Tolerance



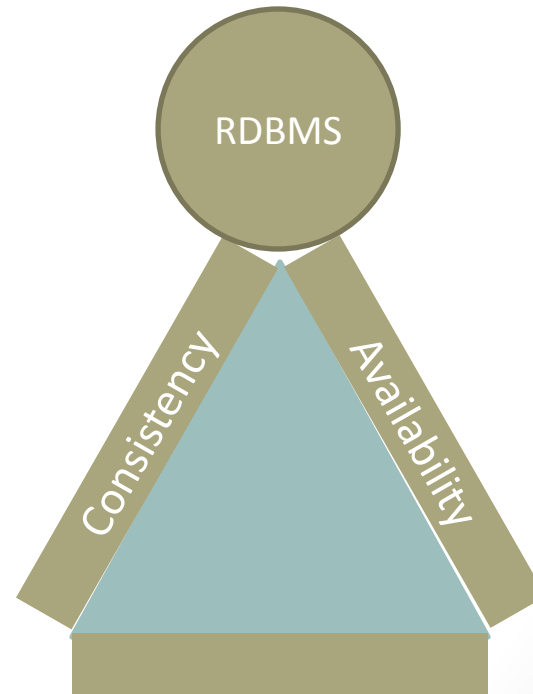
CAP Theorem

- CAP stands for:
 - Consistency: All nodes see the same data at the same time
 - Availability: Nodes are available for updates and reads
 - Partition Tolerance: Arbitrary message loss or partial failure does not bring down the system



CAP Theorem

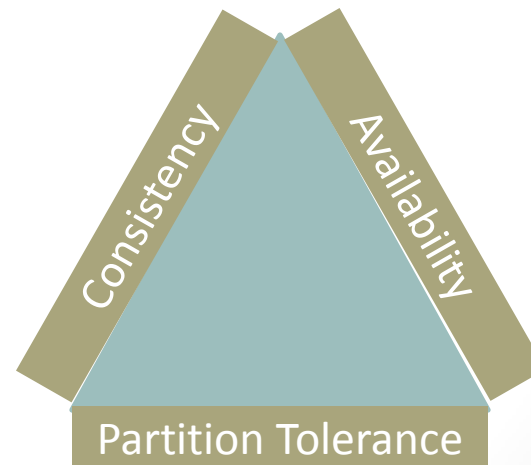
- Assume a single node with one set of data.
- This simple system resembles a typical RDBMS.
- Partition tolerance is irrelevant, because we only have one node.



CAP Theorem

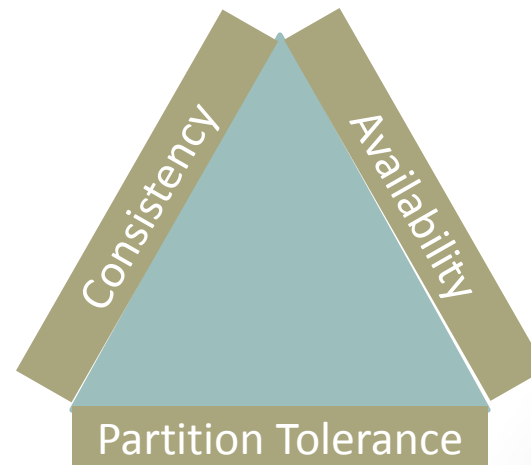
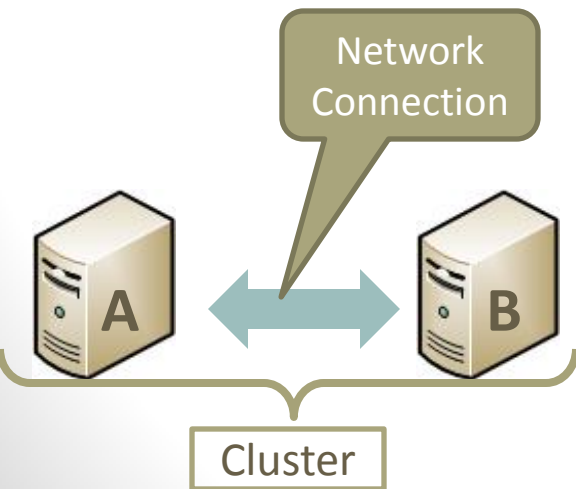


- The CAP theorem was formulated by Eric Brewer
http://en.wikipedia/wiki/CAP_theorem
- Two formulations of the CAP theorem:
 - You can have at most two of the CAP properties for any shared data system.
 - During a network partition, a distributed system must choose either Consistency or Availability.



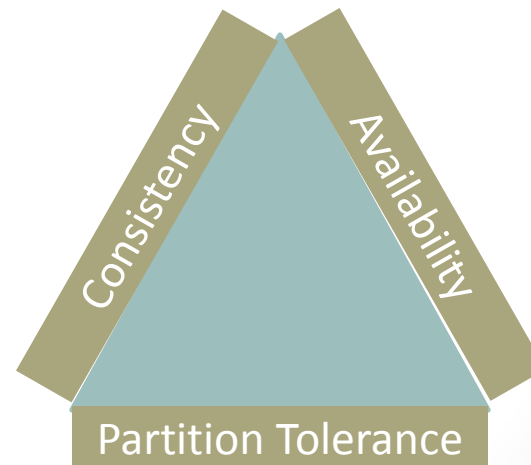
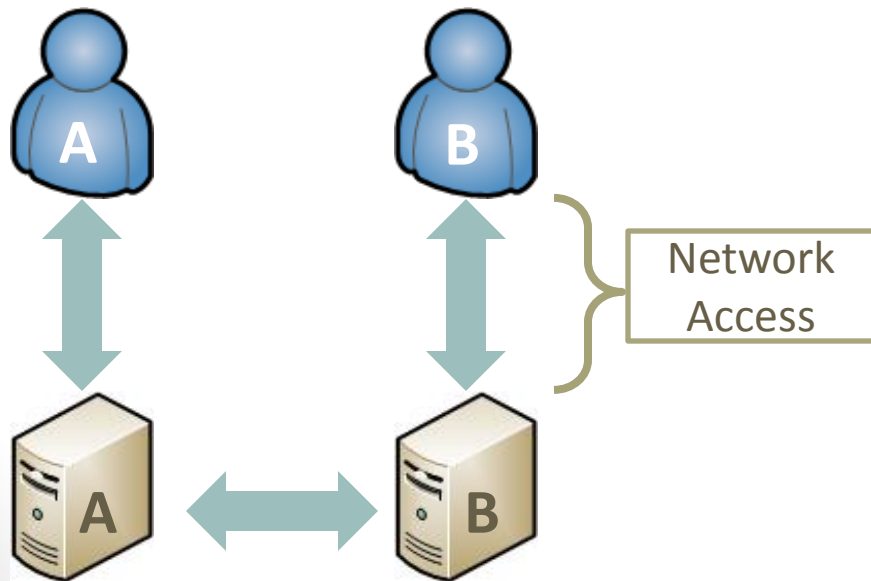
CAP Theorem

- Assume a cluster with shared and replicated data.
- The cluster consists of two connected nodes called A and B.



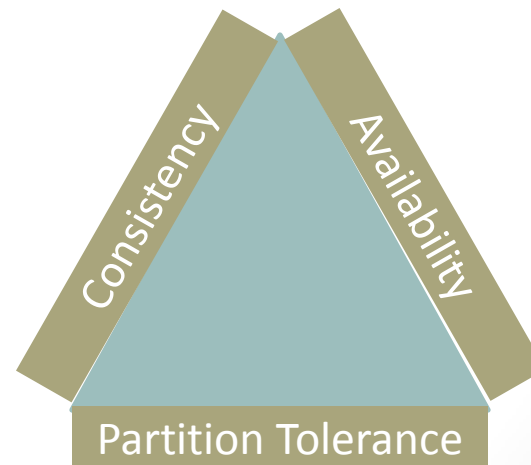
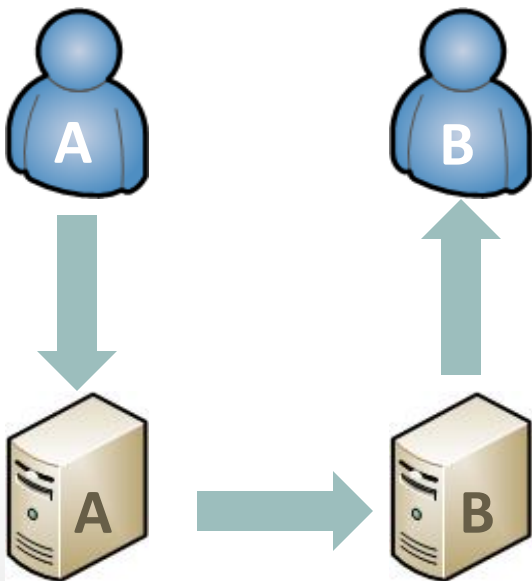
CAP Theorem

- Assume a cluster with shared and replicated data.
- The cluster consists of two connected nodes called A and B.
- The cluster is used by two users, called A and B. Each user has network access to a separate node



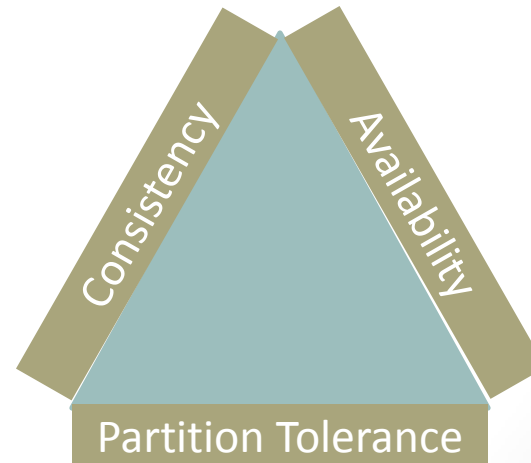
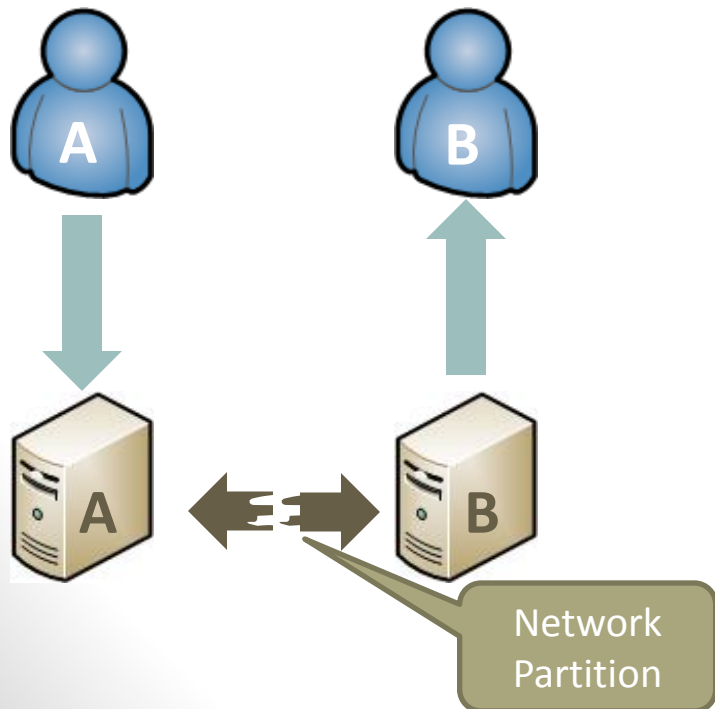
CAP Theorem

- Scenario 1: Network is available and Data are Consistent
 1. User A updates node A
 2. Update is communicated to node B
 3. User B reads the update from node B



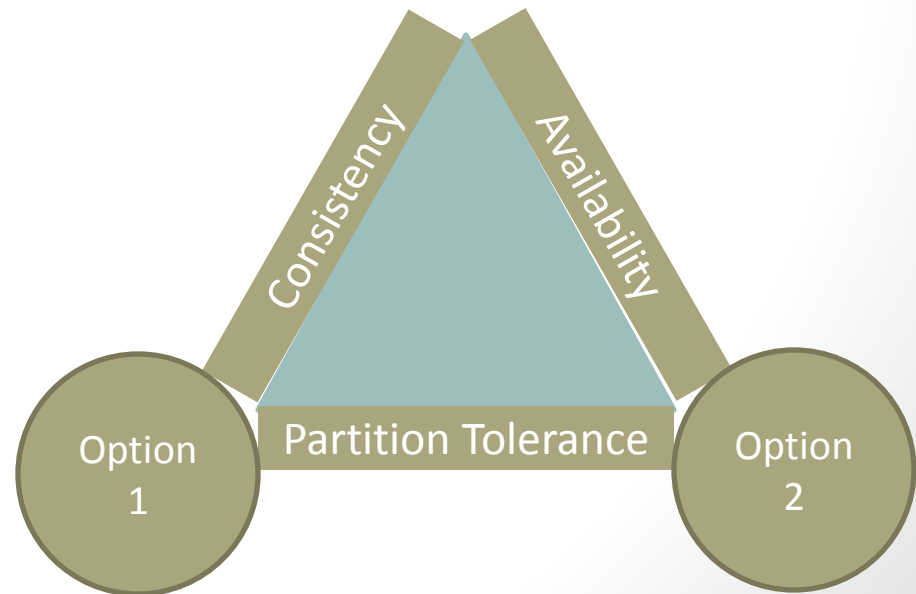
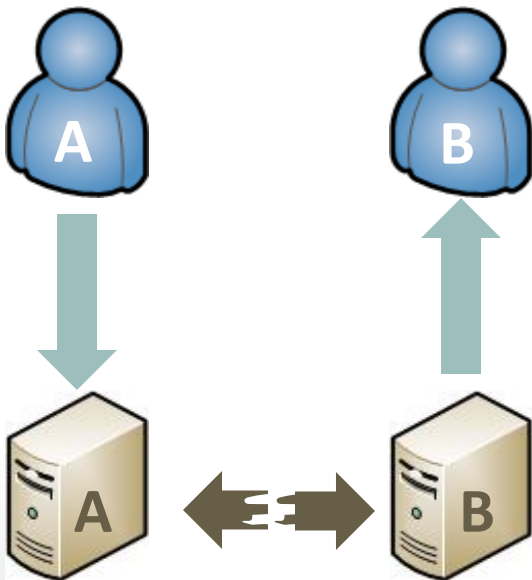
CAP Theorem

- Scenario 2: A network failure occurred.
 1. User A attempts to update node A
 2. Any Update cannot be communicated
 3. User B attempts to read the update



CAP Theorem

- Scenario 2: A network failure occurred. Two options:
 1. Make the database unavailable to avoid inconsistency
 2. Keep the database available and tolerate inconsistency



NoSQL

Assignment (1)

1. Why are performance metrics better on training data than on test data?
2. How do you determine which data are training data and which data are test data?
3. Beware, this problem contains irrelevant data while some important numbers are not explicitly presented. A model was trained on **300** individuals where **149** had the cold and **151** were healthy. The model was tested on **100** individuals where **10** were ill. The model correctly predicted that **85** of the healthy individuals were indeed healthy and correctly predicted that **7** of the ill individuals were indeed ill. The other predictions were incorrect. Consult Wikipedia: http://en.wikipedia.org/wiki/Precision_and_recall and construct a confusion matrix and then calculate the following:
 - a) Sensitivity
 - b) Specificity
 - c) Accuracy
 - d) Precision
 - e) Recall

Assignment (2)

4. The probability threshold for a classification varies in an ROC chart from 0 to 1.
 - a) What point of the graph corresponds to a threshold of zero?
 - b) What point of the graph corresponds to a threshold of one?
 - c) What point of the graph corresponds to a threshold of 0.5?
(trick question)
5. A Classification is tested on 1000 cases. In the approximate middle of its ROC chart there is a point where the false positive rate is 0.4, the true positive rate is 0.8, and the accuracy is 0.7.
 - a) What does the confusion matrix look like?
 - b) What can you say about the probability threshold at that point?
(trick question)

Assignment (3)

6. In HowToMakeAnROC.xls, complete the Exercises 1 and 2 and graph both of these ROC charts in the same Excel file.
7. Submit answers to items 1 through 5 in a text file. Submit the completed Excel file from item 6. Submission deadline is Saturday 11:00 PM.
8. Look through the new terminology at the end of this slide deck and read:
 - Google file system:
<http://static.googleusercontent.com/media/research.google.com/en/us/archive/gfs-sosp2003.pdf>
 - MapReduce:
<http://static.googleusercontent.com/media/research.google.com/en/us/archive/mapreduce-osdi04.pdf>

New Terminology

- Hadoop
- Master Node
- Data Node
- Cluster
- Hive
- Impala
- MapReduce
- HDFS
- Doug Cutting
- Scalability
- AWS
- Elastic Cloud
- NoSQL
- CAP Theorem
- Consistency (CAP)
- Availability (CAP)
- Partition Tolerance (CAP)
- Eric Brewer
- RDBMS
- ACID
- Atomic (ACID)
- Consistent (ACID)
- Isolation (ACID)
- Durability (ACID)
- BASE
- Eventual Consistency
- Paxos
- Sqoop
- CouchDB
- Shared Data
- Stale Data
- Scale-out
- Scale-up
- Grace Hopper
- Data Replication
- Horizontal Partitioning
- Vertical Partitioning
- Heartbeats
- Multi-Version Concurrency Control
- EAV
- Relational Algebra
- Relational Calculus
- Relational Model
- Ted Codd
- Codd's Theorem
- Transaction Shell
- Column-oriented DBMS
- Row-oriented
- SPARQL

Introduction to Data Science