CMSC 473/673
Natural Language Processing

Instructor: Lara J. Martin (she/they)

TA: Duong Ta (he)

Slides modified from Dr. Frank Ferraro

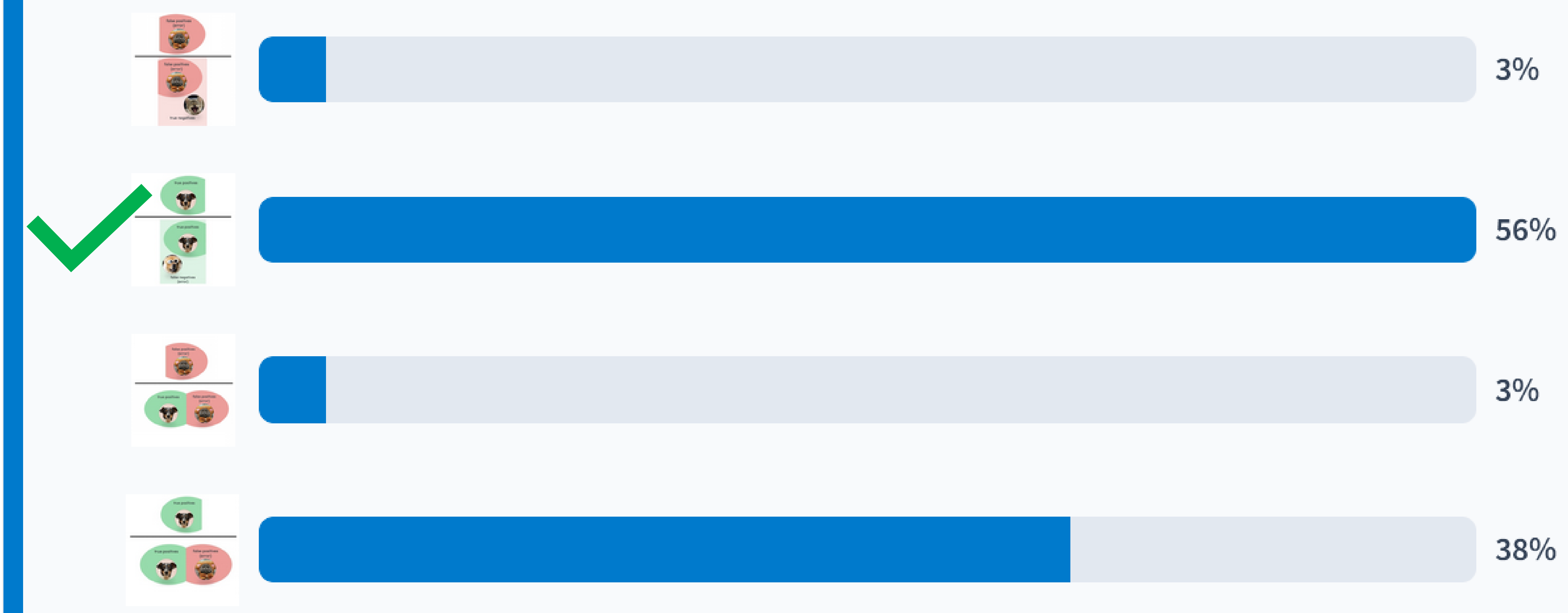
Learning Objectives

Develop an intuition about precision & recall

Extend P/R to multi-class problems

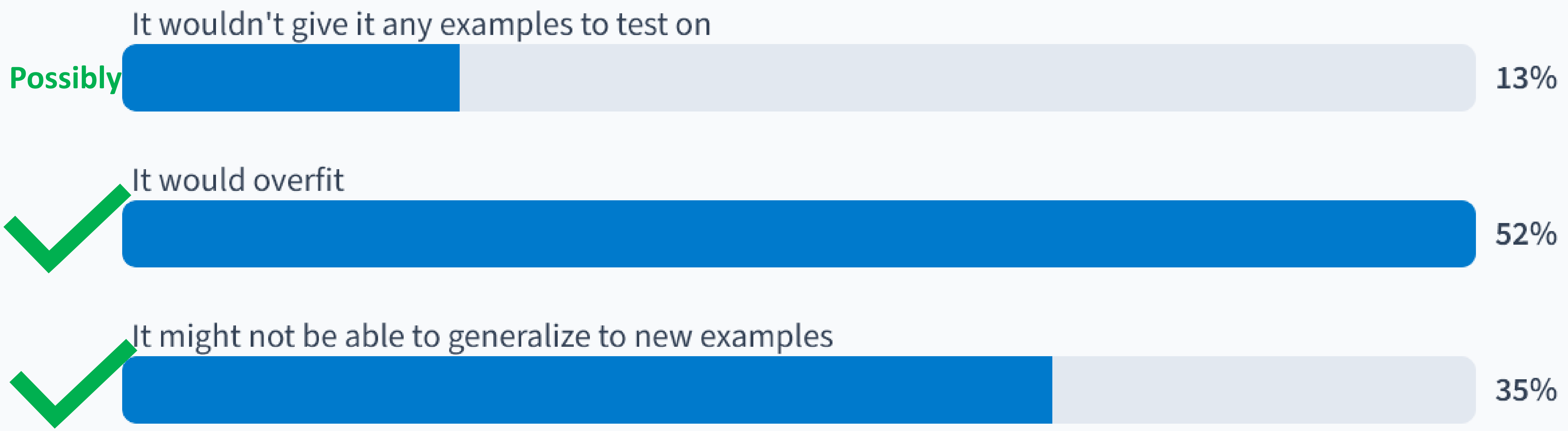
Identify when you might want certain evaluation metrics over others

If you are classifying pictures of dogs, what would be the "equation" for *recall* (where the top of the image is the numerator and the bottom of the image is the denominator)?



The difference between classification & regression is that a regression model will produce a continuous output.

Why would you want to divide up your data (instead of training on it all)?



Contingency Table (out of table form)

Query:
Articles about dogs

FN

Adopting an Animal at the Pound

Why Huskies Howl

Who's a good boy?

TP

10 Tips for Grooming your Dog

What Foods to Hide from Your Dog During the Holidays

TN

15 Weird Animals You Can Find in North America

When Cats Rule the World

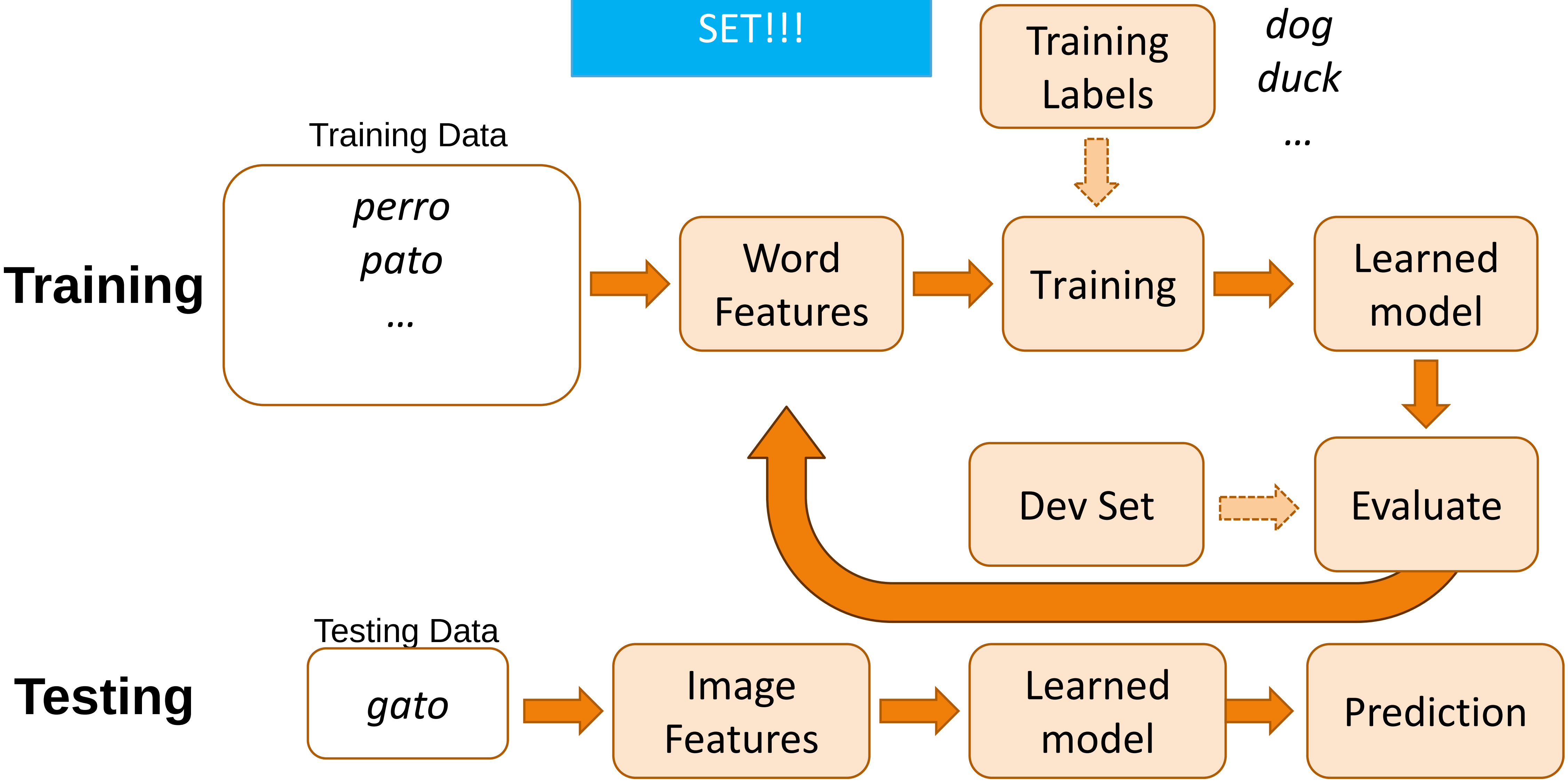
FP

It's raining cats and dogs

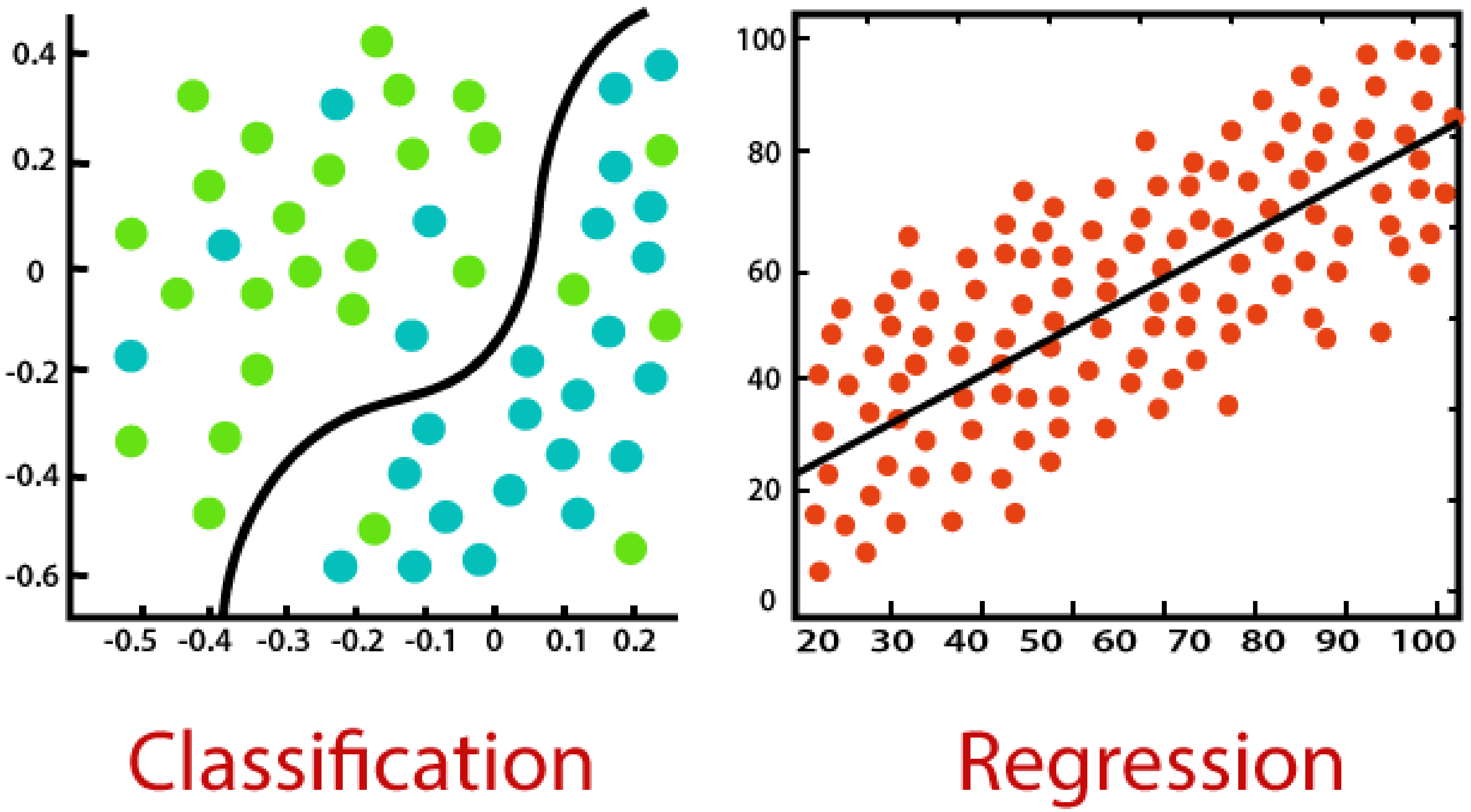
Meme from: *https://www.reddit.com/r/AdviceAnimals/comments/ck8xh0/yo_dawg_i_heard_you_like_old_memes/*

Review: Steps

DO NOT ITERATE ON THE TESTING SET!!!



Review: Types of models



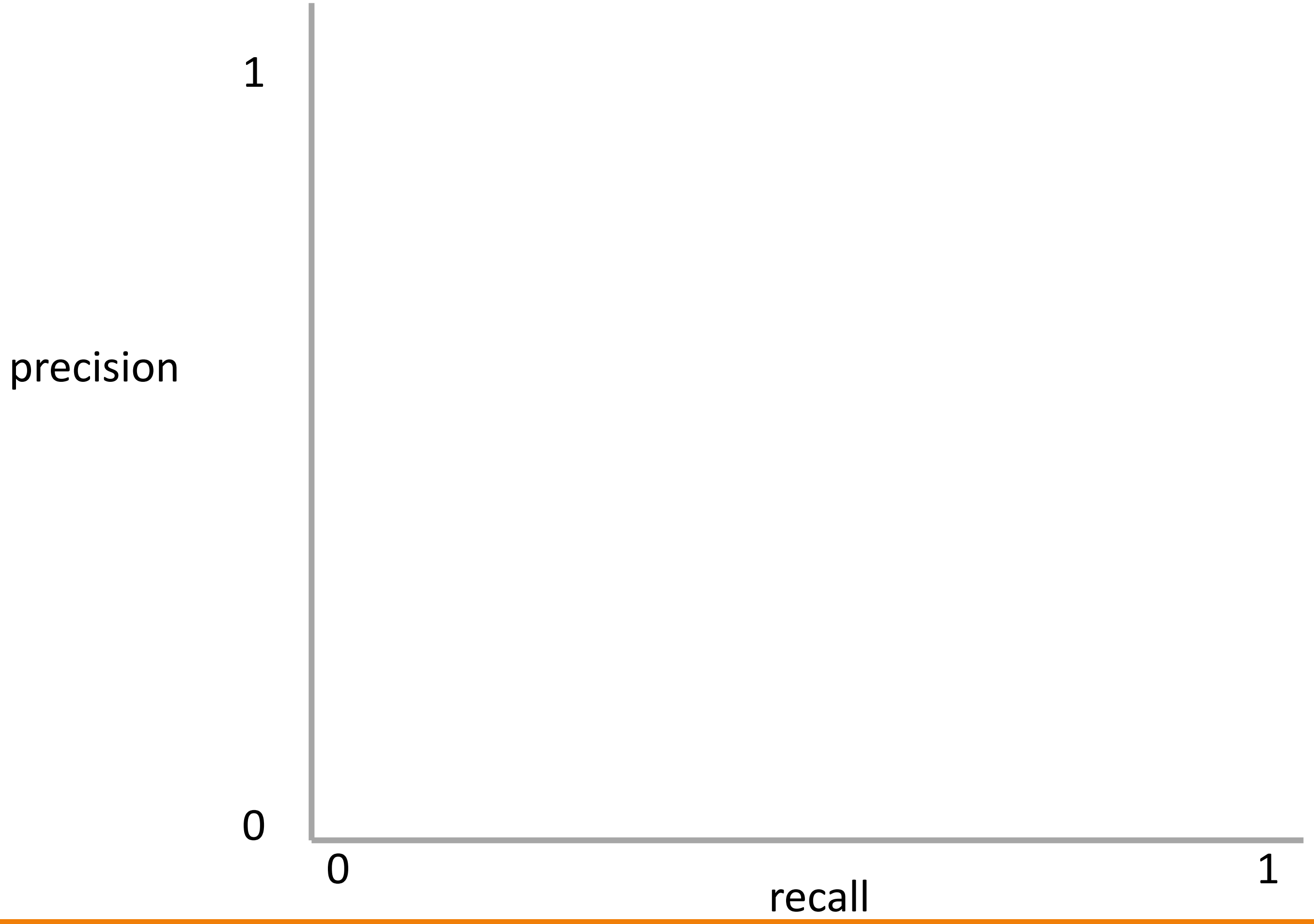
https://medium.com/unpackai/classification-regression-in-machine-learning-7cf3b13b0b09

Review: Classification Evaluation: the 2-by-2 contingency table

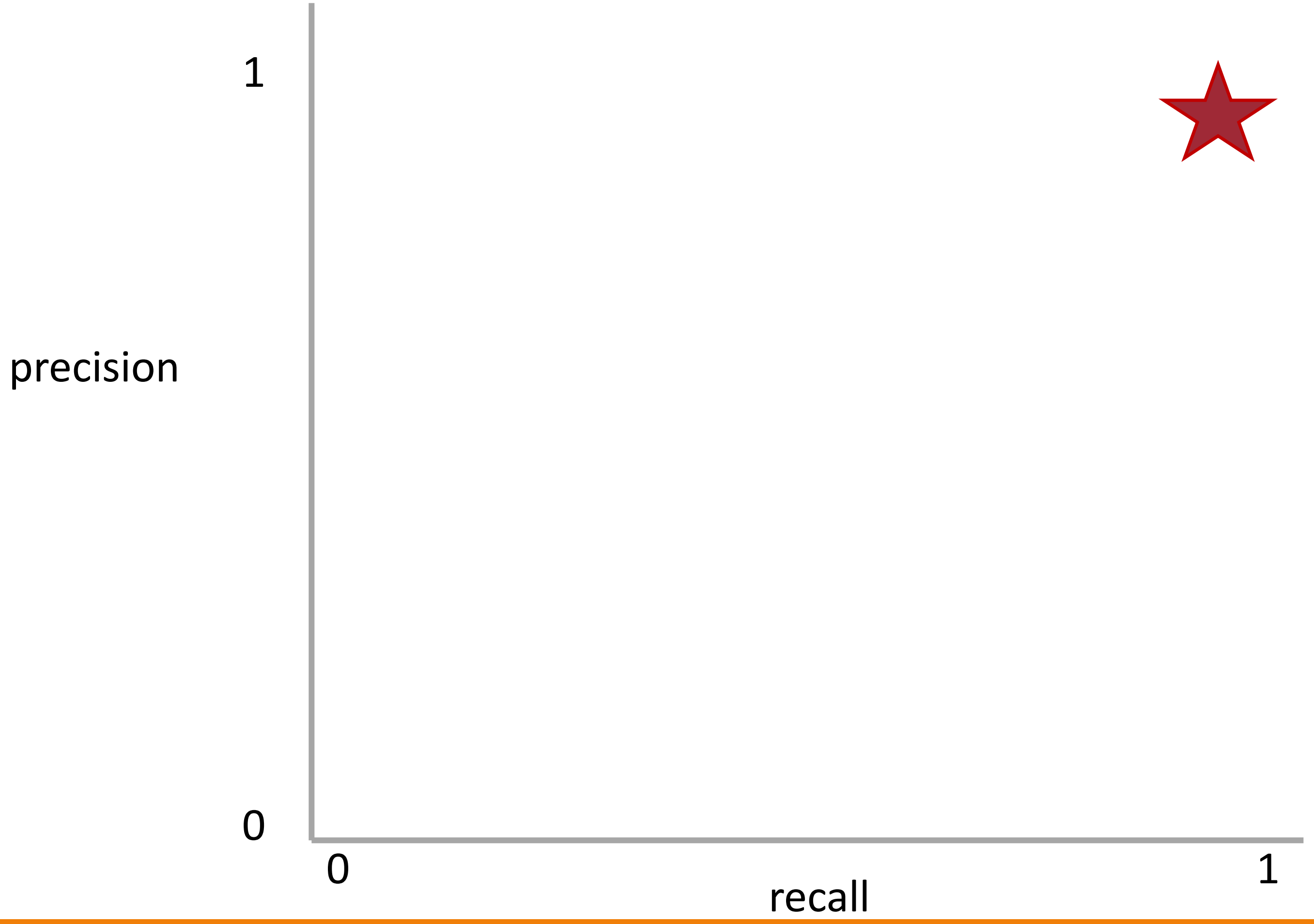
	What is the actual label?	
What label does our system predict? (↓)	Actual Target Class ("●")	Not Target Class ("○")
Selected/ Guessed ("●")	True Positive (TP) ● *Actual* ● *Guessed*	False Positive (FP) ○ *Actual* ● *Guessed*
Not selected/ not guessed ("○")	False Negative (FN) ● *Actual* ○ *Guessed*	True Negative (TN) ○ *Actual* ○ *Guessed*

Precision and Recall Present a Tradeoff

Q: Where do you want your ideal model?



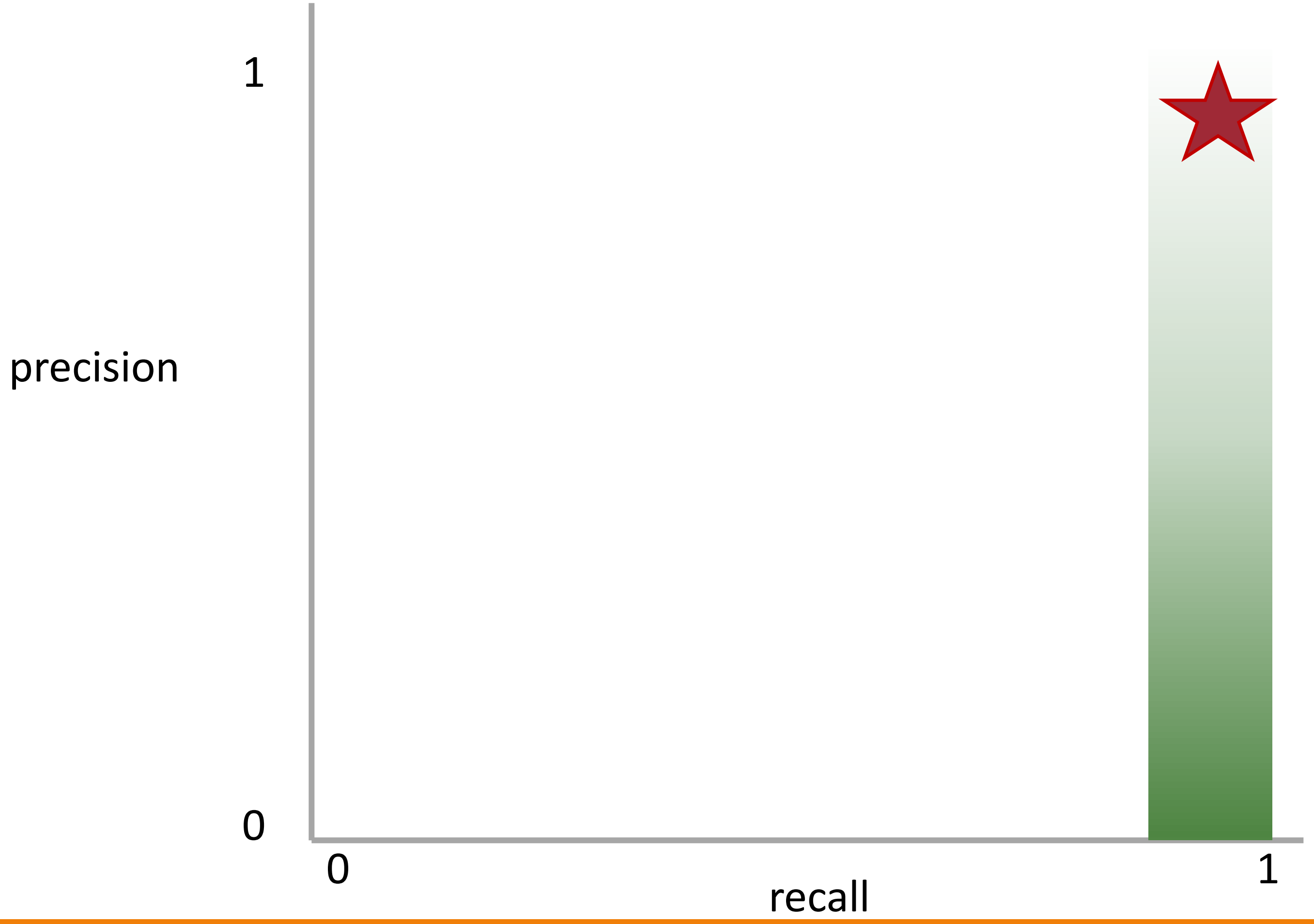
Precision and Recall Present a Tradeoff



Q: Where do you want your ideal model?

Q: You have a model that always identifies correct instances. Where on this graph is it?

Precision and Recall Present a Tradeoff

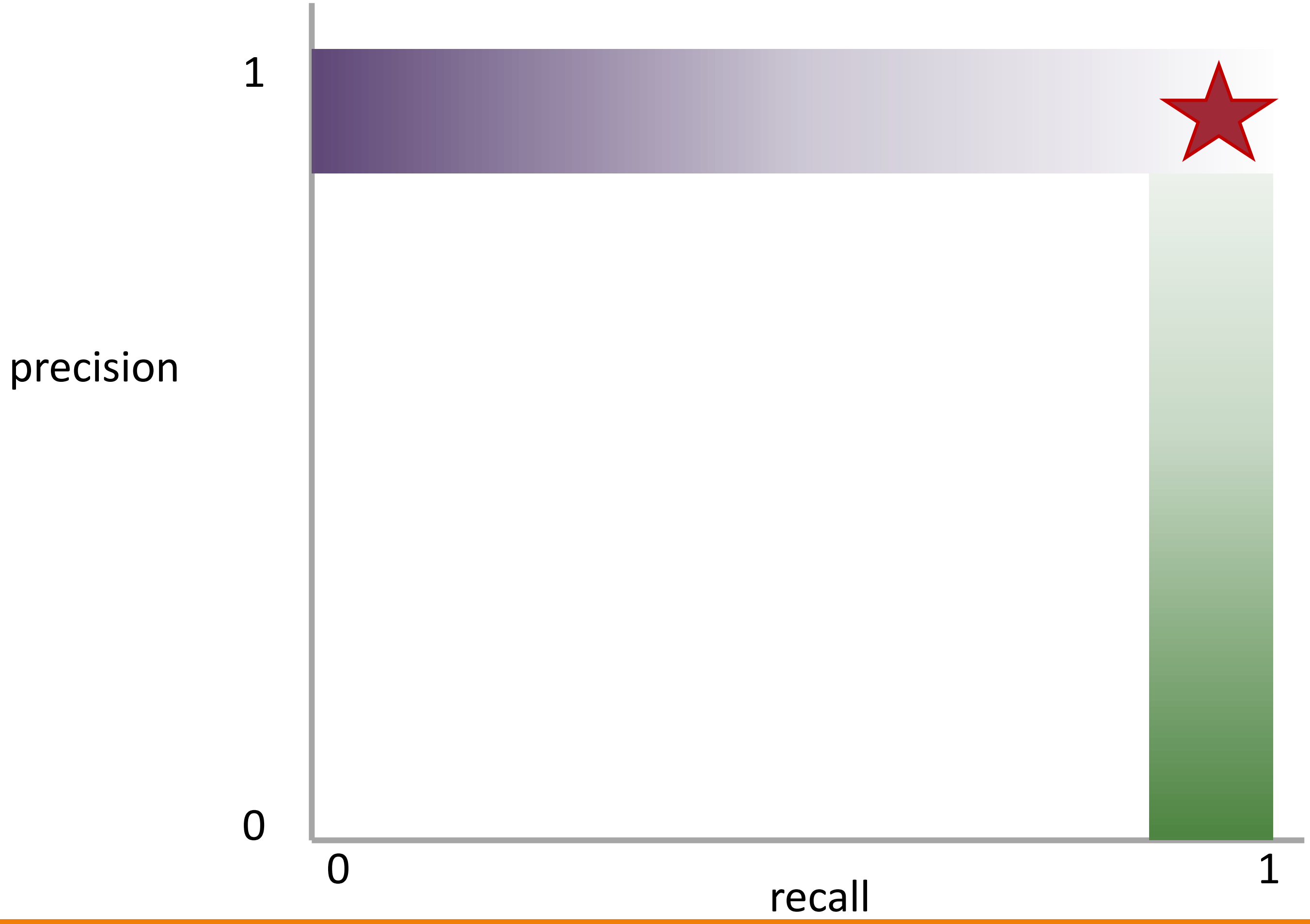


Q: Where do you want your ideal model ?

Q: You have a model that always identifies correct instances. Where on this graph is it?

Q: You have a model that only make correct predictions. Where on this graph is it?

Precision and Recall Present a Tradeoff

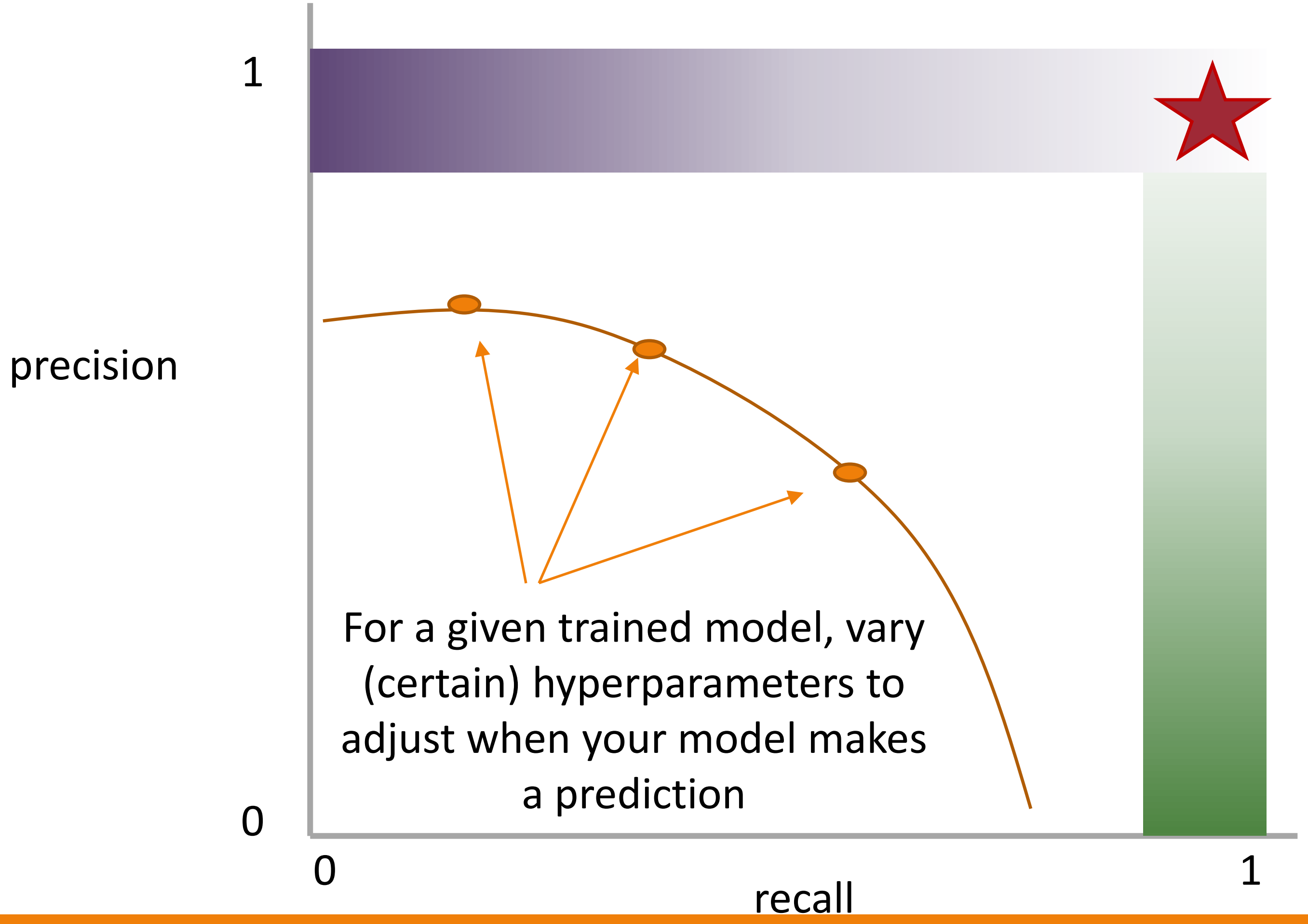


Q: Where do you want your ideal model?

Q: You have a model that always identifies correct instances. Where on this graph is it?

Q: You have a model that only make correct predictions. Where on this graph is it?

Precision and Recall Present a Tradeoff



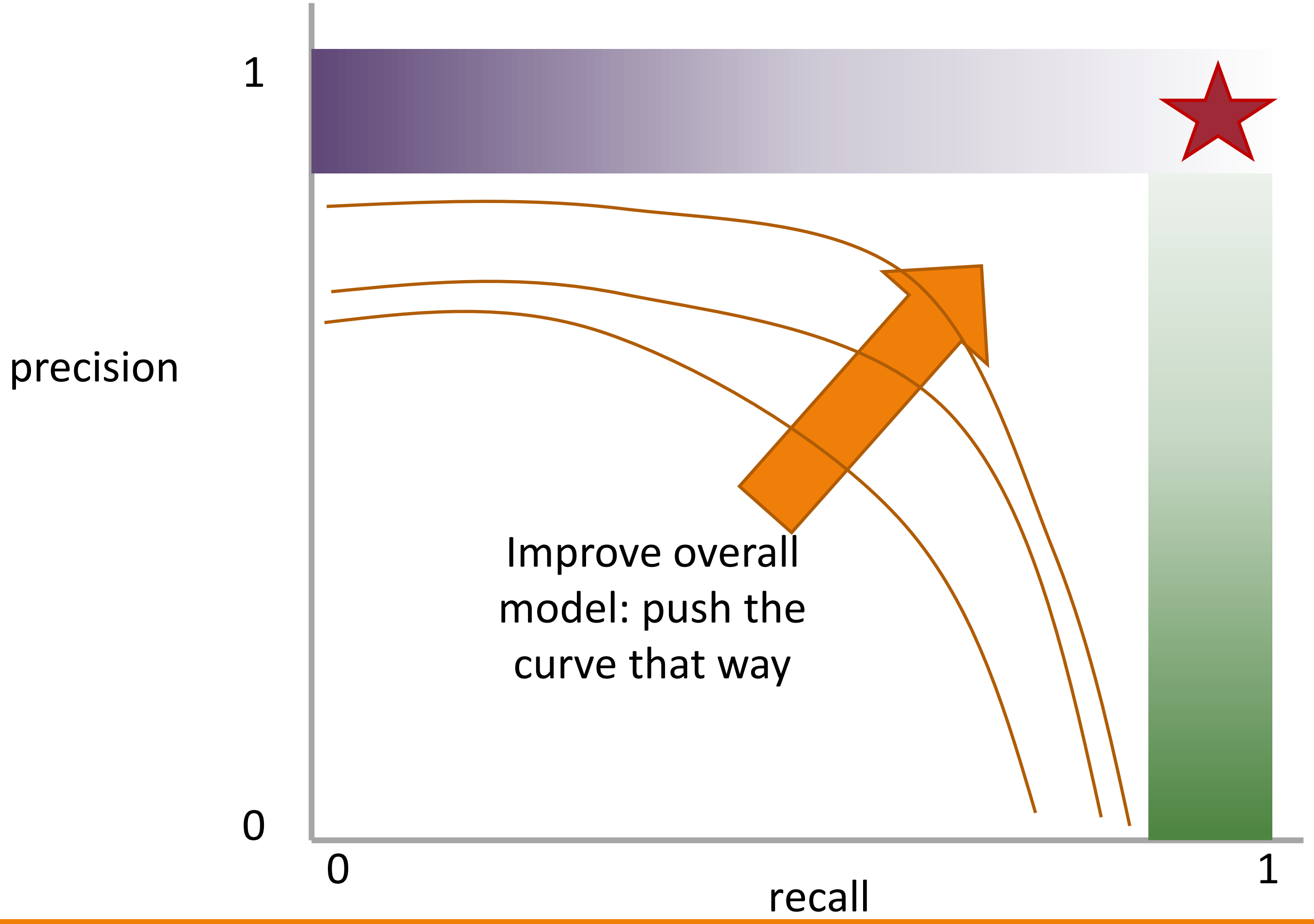
Q: Where do you want your ideal model ?

Q: You have a model that always identifies correct instances. Where on this graph is it?

Q: You have a model that only make correct predictions. Where on this graph is it?

Idea: measure the tradeoff between precision and recall

Precision and Recall Present a Tradeoff



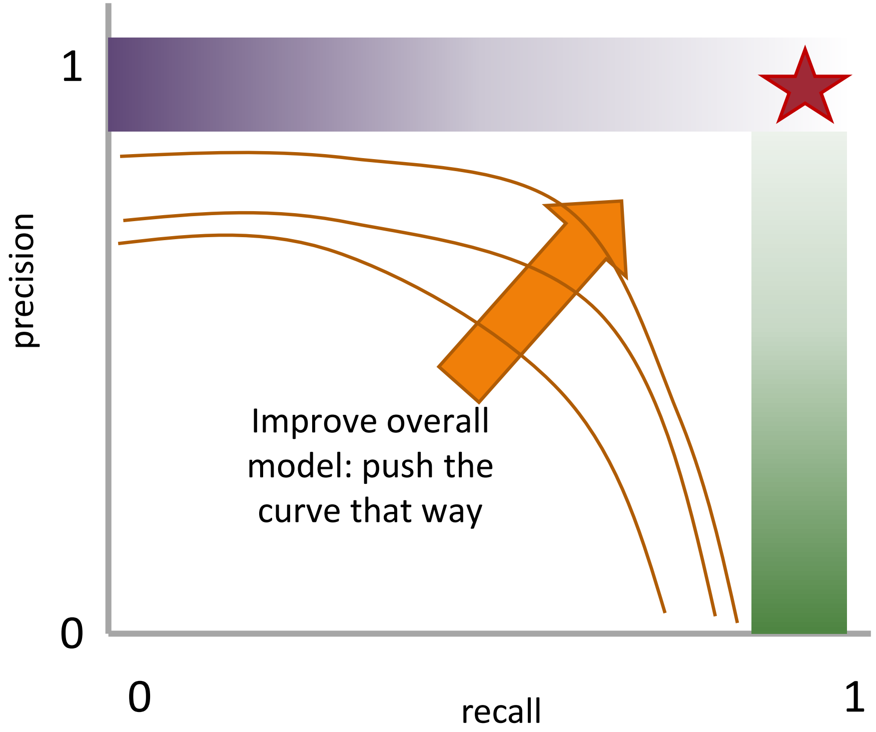
Q: Where do you want your ideal model ?

Q: You have a model that always identifies correct instances. Where on this graph is it?

Q: You have a model that only make correct predictions. Where on this graph is it?

Idea: measure the tradeoff between precision and recall

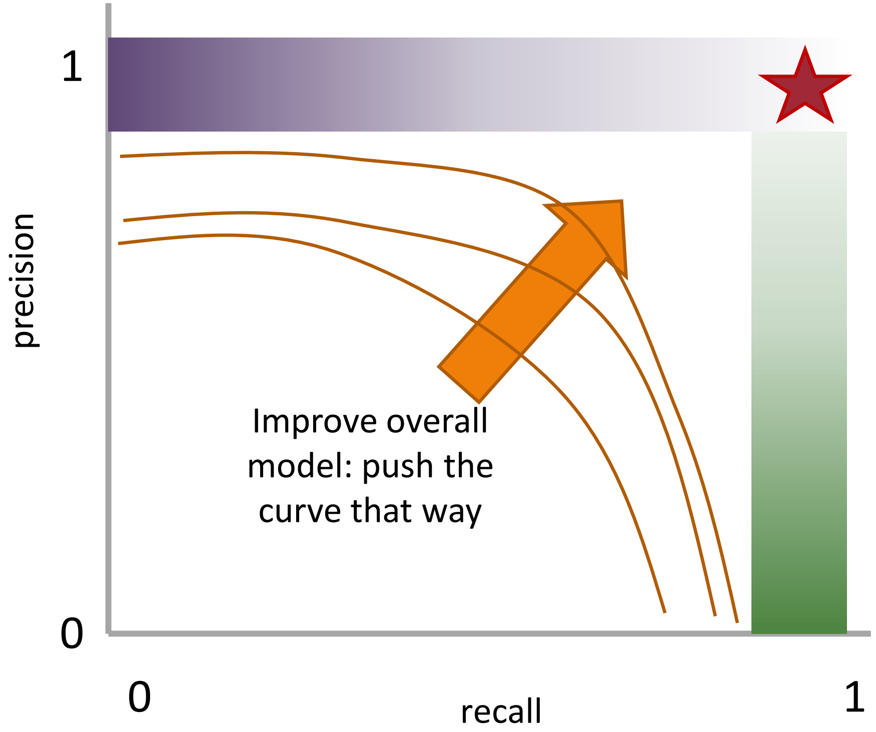
Measure this Tradeoff: Area Under the Curve (AUC)



AUC measures the area under this tradeoff curve

Min AUC: 0 ☹
Max AUC: 1 ☺

Measure this Tradeoff: Area Under the Curve (AUC)



Min AUC: 0 ☹
Max AUC: 1 ☺

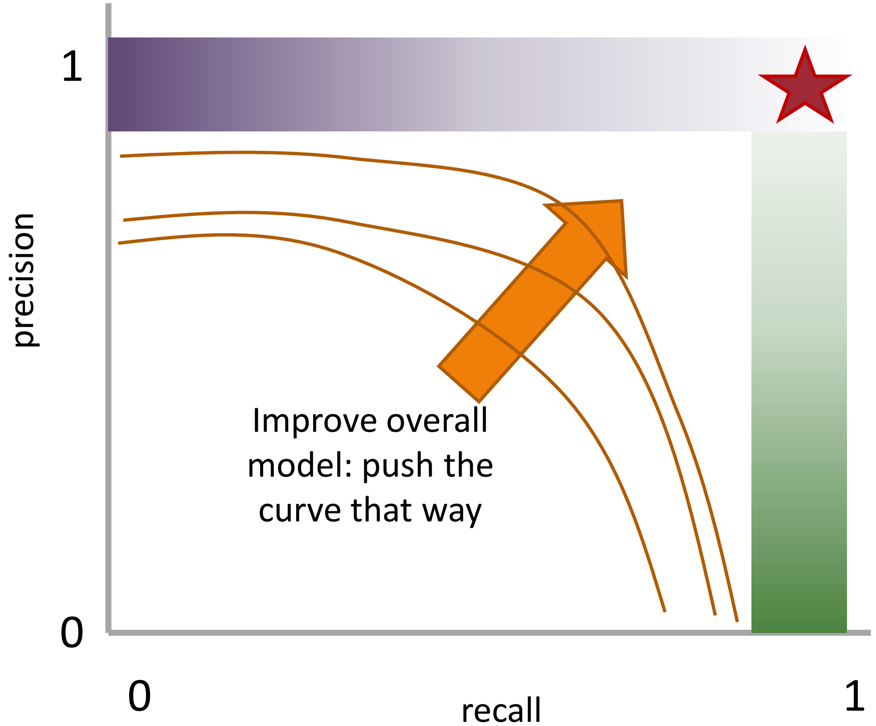
AUC measures the area under this tradeoff curve

1. Computing the curve

   You need true labels & predicted labels with some score/confidence estimate

   Threshold the scores and for each threshold compute precision and recall

Measure this Tradeoff: Area Under the Curve (AUC)



Min AUC: 0 ☹
Max AUC: 1 ☺

AUC measures the area under this tradeoff curve

1. Computing the curve

   You need true labels & predicted labels with some score/confidence estimate

   Threshold the scores and for each threshold compute precision and recall

2. Finding the area

   How to implement: trapezoidal rule (& others)

In practice: external library like the sklearn.metrics module

A combined measure: F-score

Weighted (harmonic) average of **P**recision & **R**ecall

F1 measure: equal weighting between precision and recall

$$F_1 = \frac{2 \ * P \ * R}{P + R}$$

A combined measure: F-score

Weighted (harmonic) average of **P**recision & **R**ecall

F1 measure: equal weighting between precision and recall

$$F_1 = \frac{2 * P * R}{P + R} = \frac{2 * TP}{2 * TP + FP + FN}$$

(useful when $P = R = 0$)

Classification Evaluation: Accuracy, Precision, and Recall

Accuracy: % of items correct

$$\frac{\mathrm{TP} + \mathrm{TN}}{\mathrm{TP} + \mathrm{FP} + \mathrm{FN} + \mathrm{TN}}$$

$$F_1 = \frac{2 * P * R}{P + R} = \frac{2 * TP}{2 * TP + FP + FN}$$

Accuracy takes everything in consideration

F-Score is focused on TP

When would you want to use accuracy vs F1?

Accuracy works better if the dataset is balanced

	Actually Target	Actually Not Target
Selected/Guessed	True Positive (TP)	False Positive (FP)
Not select/not guessed	False Negative (FN)	True Negative (TN)

Implementation: How To

1. scikit-learn: sklearn.metrics
   - very stable

2. huggingface evaluate module
   - community input
   - sometimes are based on sklearn

3. implement your own

P/R/F in a Multi-class Setting: Micro- vs. Macro-Averaging

If we have more than one class, how do we combine multiple performance measures into one quantity?

Macroaveraging: Compute performance for each class, then average.

Microaveraging: Collect decisions for all classes, compute contingency table, evaluate.

P/R/F in a Multi-class Setting: Micro- vs. Macro-Averaging

Macroaveraging: Compute performance for each class, then average.

$$\text{macroprecision} = \frac{1}{C}\sum_{c} \frac{\mathrm{TP_c}}{\mathrm{TP_c} + \mathrm{FP_c}} = \frac{1}{C}\sum_{c} \text{precision}_c$$

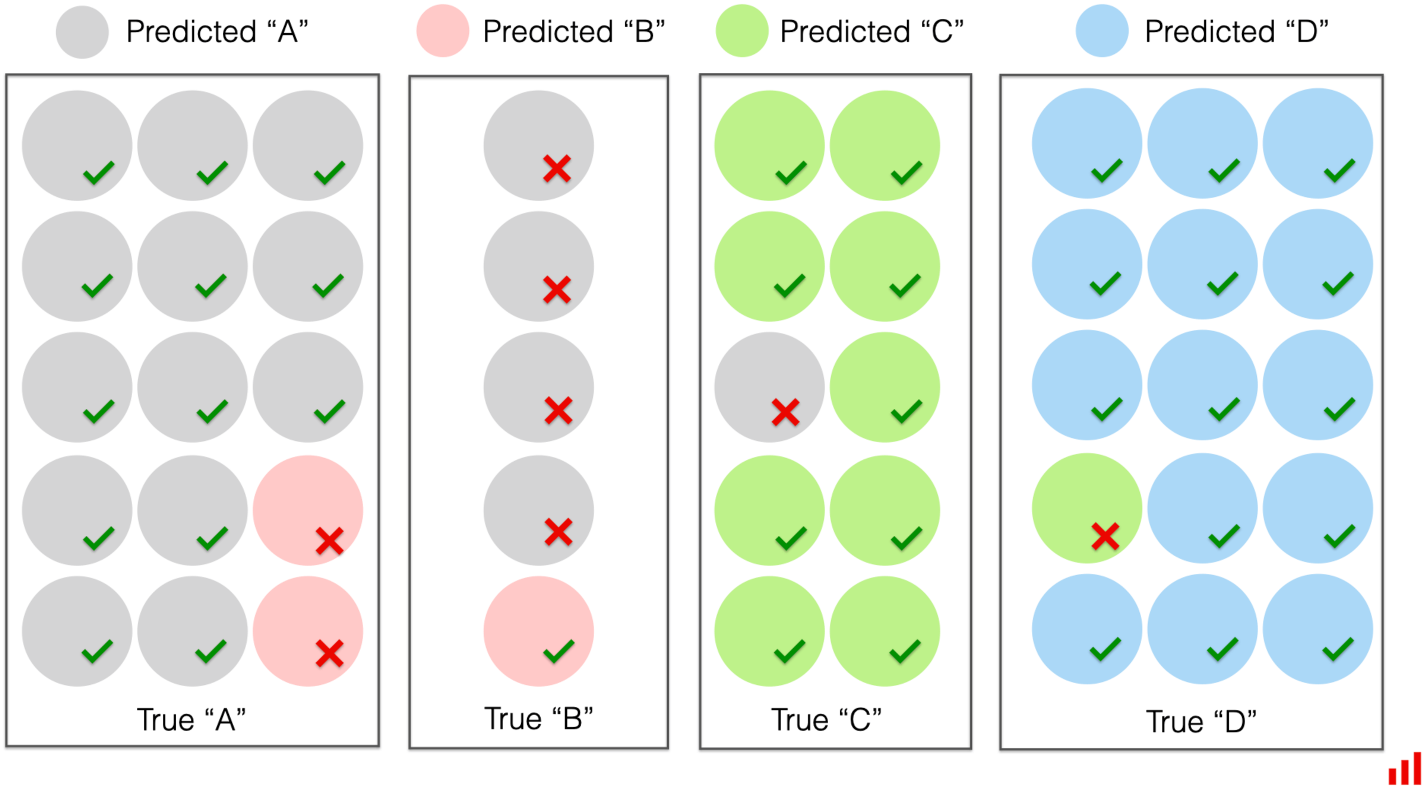
$$\text{macrorecall} = \frac{1}{C}\sum_{c} \frac{\mathrm{TP_c}}{\mathrm{TP_c} + \mathrm{FN_c}} = \frac{1}{C}\sum_{c} \text{recall}_\mathrm{c}$$

Microaveraging: Collect decisions for all classes, compute contingency table, evaluate.

$$\text{microprecision} = \frac{\sum_\mathrm{c} \mathrm{TP_c}}{\sum_\mathrm{c} \mathrm{TP_c} + \sum_\mathrm{c} \mathrm{FP_c}}$$

$$\text{microrecall} = \frac{\sum_\mathrm{c} \mathrm{TP_c}}{\sum_\mathrm{c} \mathrm{TP_c} + \sum_\mathrm{c} \mathrm{FN_c}}$$

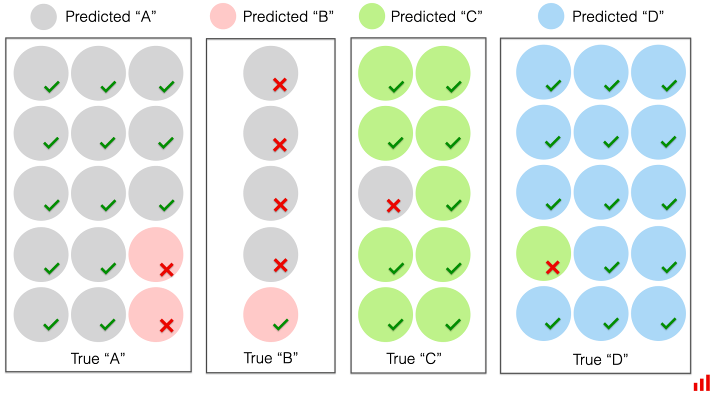
Macro/Micro Example

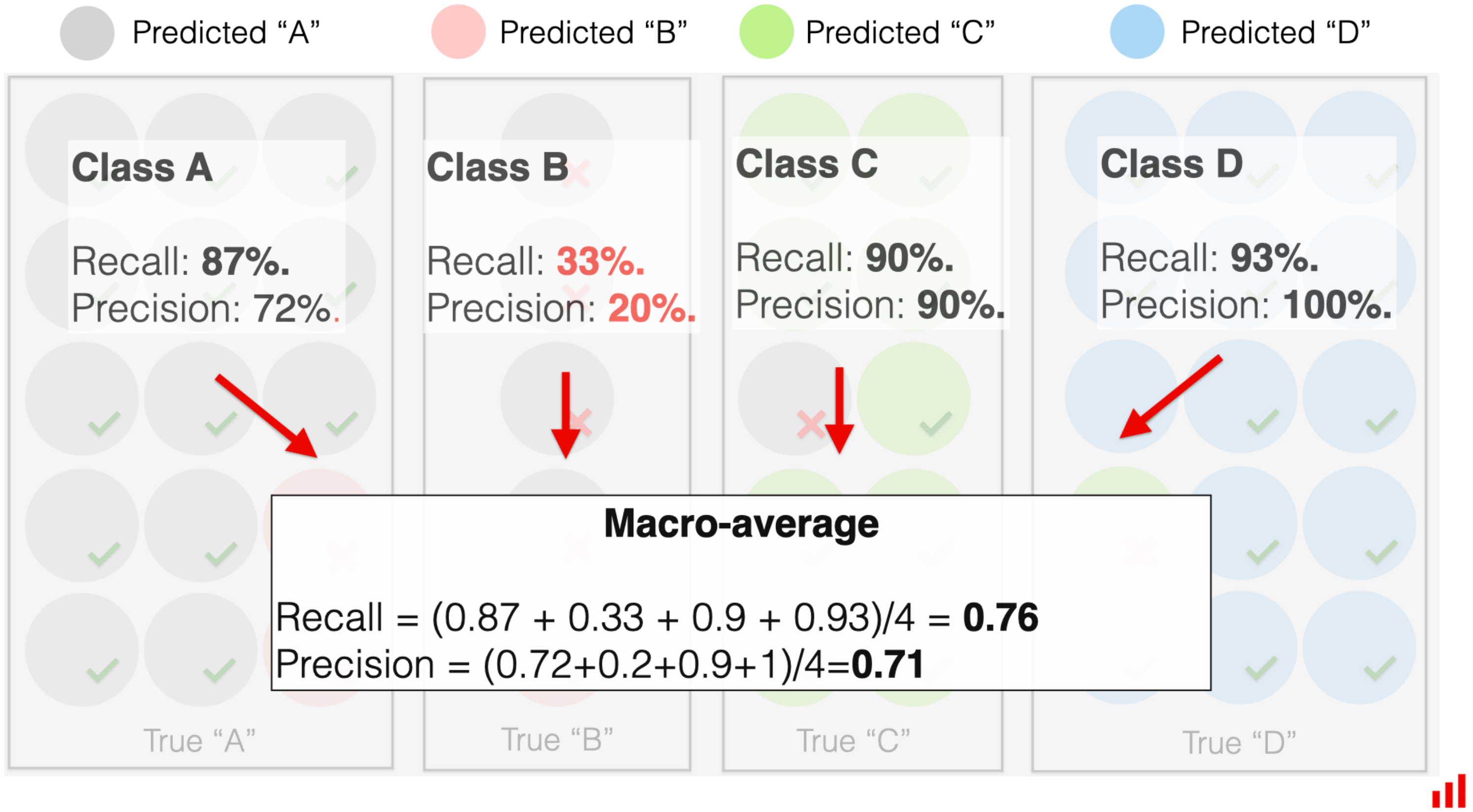


https://www.evidentlyai.com/classification-metrics/multi-class-metrics

Each class has equal weight

Macro-Average



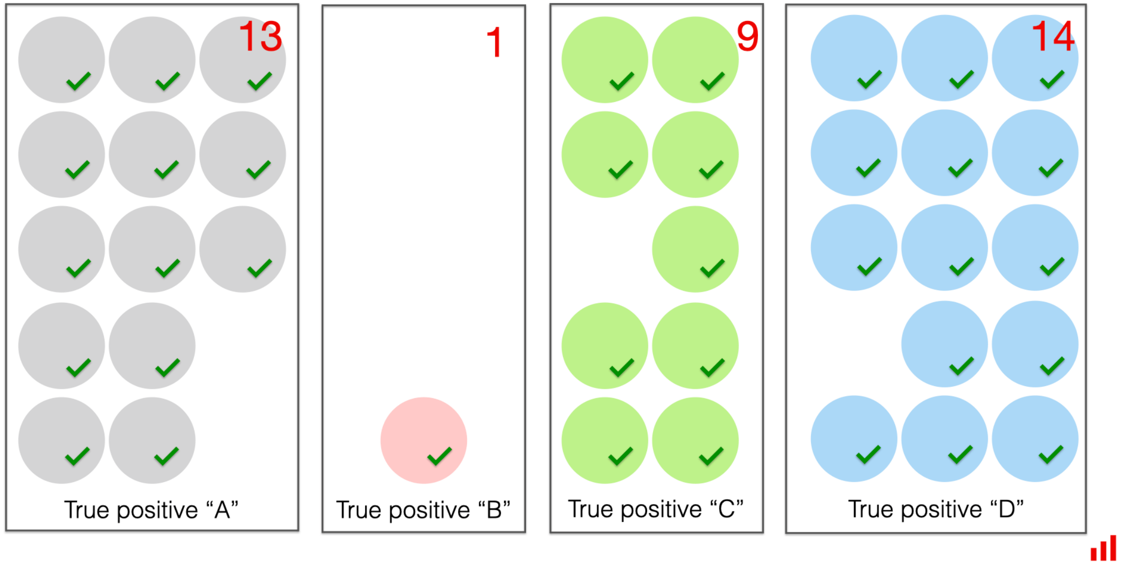


https://www.evidentlyai.com/classification-metrics/multi-class-metrics

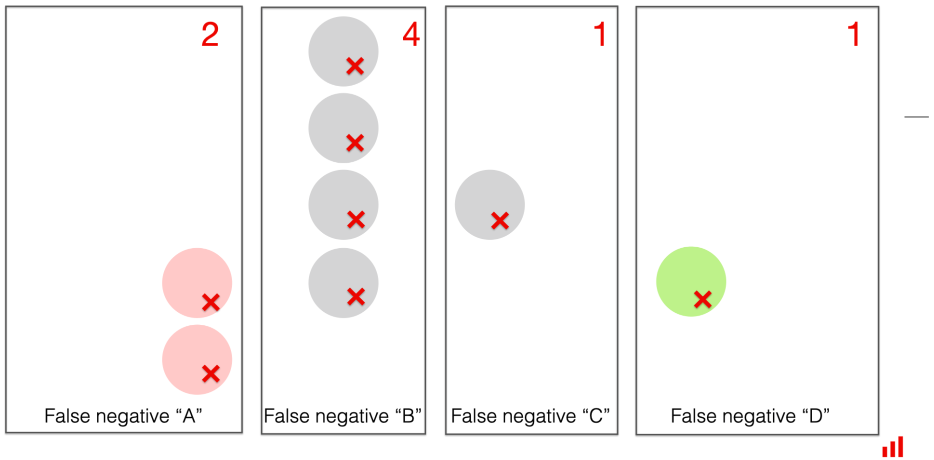
Each *instance* has equal weight

Micro-Average

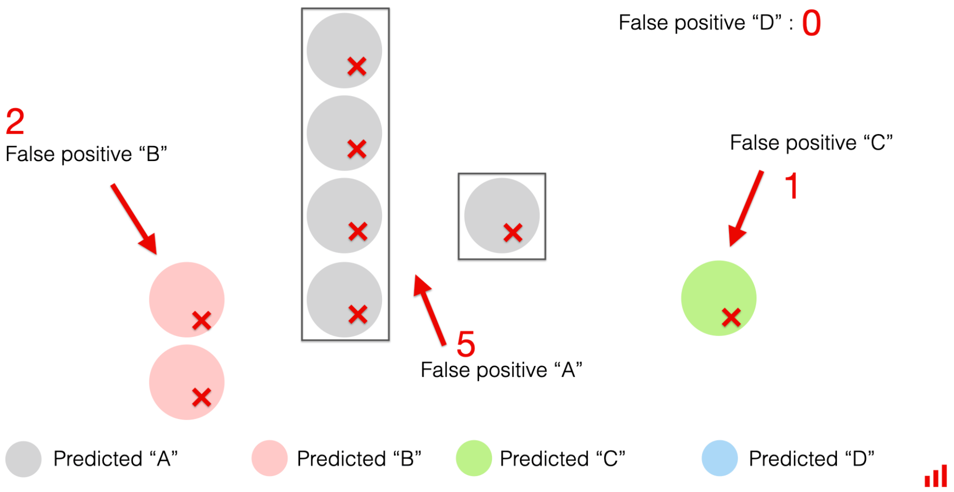
All true positives



All false negatives



All false positives



Total TP   Total FP   Total FN

$$\text{Precision}_{\text{Micro-average}} = \frac{13+1+9+14}{13+1+9+14 + \boxed{2+5+1+0}} = 0.82$$

$$\text{Recall}_{\text{Micro-average}} = \frac{13+1+9+14}{13+1+9+14 + 2+4+1+1} = 0.82$$

https://www.evidentlyai.com/classification-metrics/multi-class-metrics

Micro- vs Macro-Average

So when would we want to prefer micro-averaging vs macro-averaging?

$$\text{macroprecision} = \frac{1}{C}\sum_{c} \frac{\mathrm{TP_c}}{\mathrm{TP_c} + \mathrm{FP_c}} = \frac{1}{C}\sum_{c} \text{precision}_c$$

$$\text{macrorecall} = \frac{1}{C}\sum_{c} \frac{\mathrm{TP_c}}{\mathrm{TP_c} + \mathrm{FN_c}} = \frac{1}{C}\sum_{c} \text{recall}_\mathrm{c}$$

$$\text{microprecision} = \frac{\sum_\mathrm{c} \mathrm{TP_c}}{\sum_\mathrm{c} \mathrm{TP_c} + \sum_\mathrm{c} \mathrm{FP_c}}$$

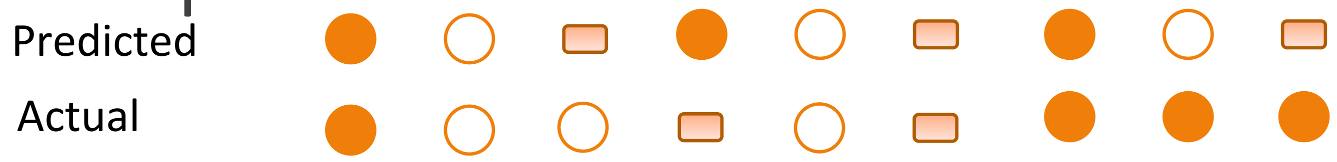
$$\text{microrecall} = \frac{\sum_\mathrm{c} \mathrm{TP_c}}{\sum_\mathrm{c} \mathrm{TP_c} + \sum_\mathrm{c} \mathrm{FN_c}}$$

But how do we compute stats for multiple classes?

Either:

1. Compute “one-vs-all” 2x2 tables. OR
2. Generalize the 2x2 tables and compute per-class TP / FP / FN based on the diagonals and off-diagonals

1. Compute “one-vs-all” 2x2 tables



Look for	Actually Target	Actually Not Target
Selected/Guessed	True Positive (TP)	False Positive (FP)
Not select/not guessed	False Negative (FN)	True Negative (TN)

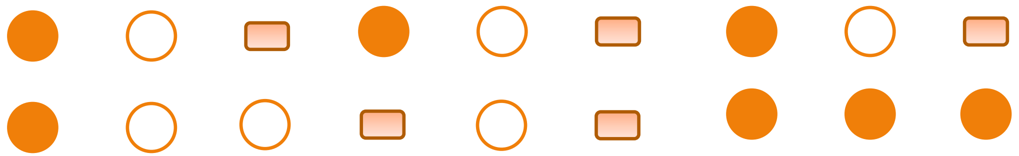
Look for	Actually Target	Actually Not Target
Selected/Guessed	True Positive (TP)	False Positive (FP)
Not select/not guessed	False Negative (FN)	True Negative (TN)

Look for	Actually Target	Actually Not Target
Selected/Guessed	True Positive (TP)	False Positive (FP)
Not select/not guessed	False Negative (FN)	True Negative (TN)

1. Compute "one-vs-all" 2x2 tables

Predicted

Actual



Look for	Actually Target	Actually Not Target
Selected/Guessed	2	1
Not select/not guessed	2	4

Look for	Actually Target	Actually Not Target
Selected/Guessed	2	1
Not select/not guessed	1	5

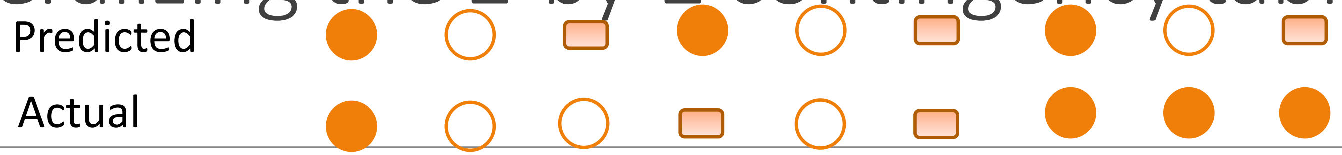
Look for	Actually Target	Actually Not Target
Selected/Guessed	1	2
Not select/not guessed	1	5

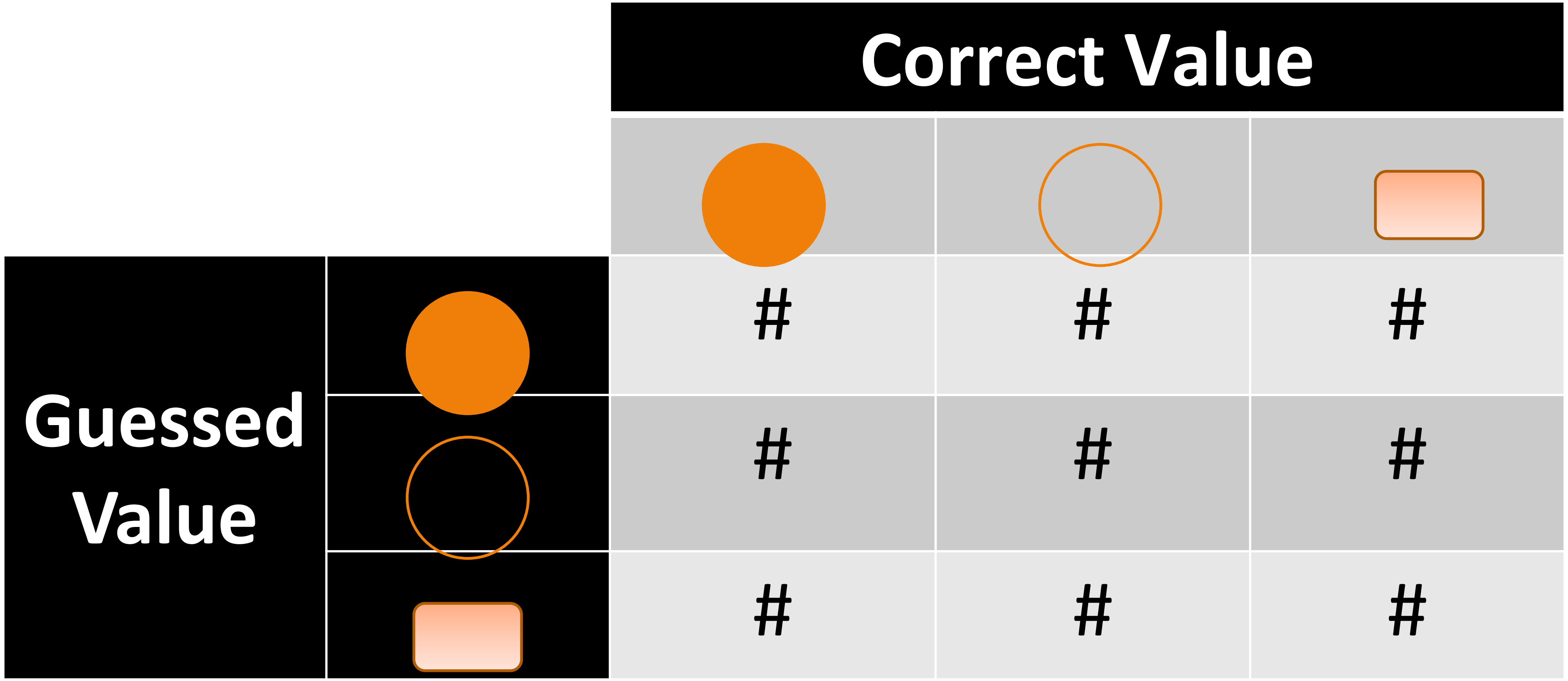
2. Generalizing the 2-by-2 contingency table

		Correct Value		
Guessed Value		#	#	#
		#	#	#
		#	#	#

This is also called a **Confusion Matrix**

2. Generalizing the 2-by-2 contingency table

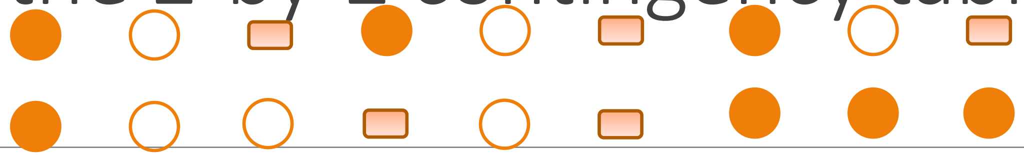




2. Generalizing the 2-by-2 contingency table

Predicted

Actual

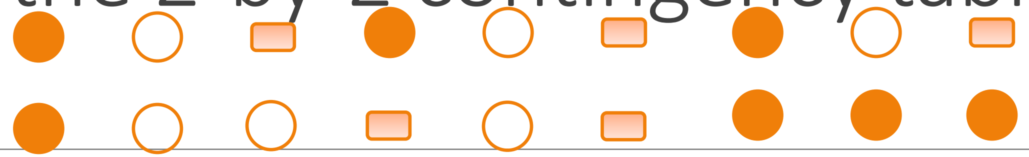


		Correct Value		
Guessed Value		2	0	1
		1	2	0
		1	1	1

2. Generalizing the 2-by-2 contingency table

Predicted

Actual



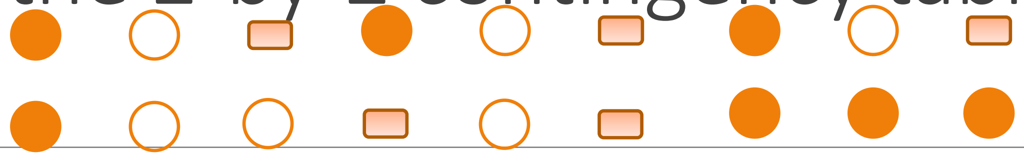
		Correct Value		
Guessed Value		A 2	B 0	C 1
		D 1	E 2	F 0
		G 1	H 1	I 1

How do you compute TP ?

2. Generalizing the 2-by-2 contingency table

Predicted

Actual



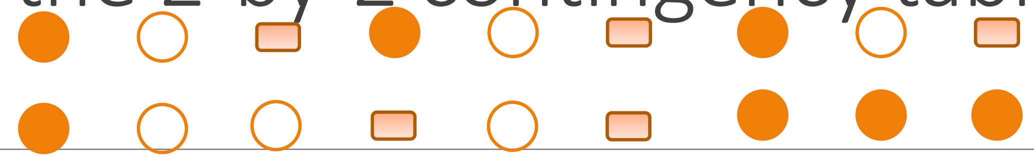
		Correct Value		
Guessed Value		A 2	B 0	C 1
		D 1	E 2	F 0
		G 1	H 1	I 1

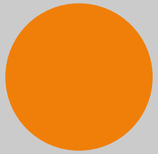
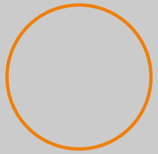
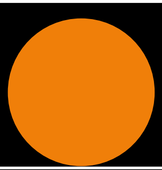
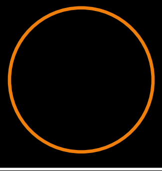
How do you compute TP ?

2. Generalizing the 2-by-2 contingency table

Predicted

Actual



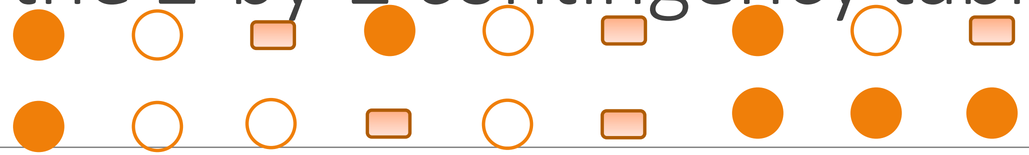
		Correct Value		
				
Guessed Value		A 2	B 0	C 1
		D 1	E 2	F 0
		G 1	H 1	I 1

How do you compute FN?

2. Generalizing the 2-by-2 contingency table

Predicted

Actual



		Correct Value		
Guessed Value		A 2	B 0	C 1
		D 1	E 2	F 0
		G 1	H 1	I 1

How do you compute FN ?

2. Generalizing the 2-by-2 contingency table

Predicted	●	○	▭	●	○	▭	●	○	▭
Actual	●	○	○	▭	○	▭	●	●	●

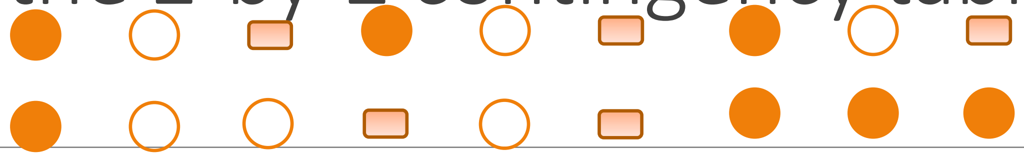
		Correct Value		
		●	○	▭
Guessed Value	●	A 2	B 0	C 1
	○	D 1	E 2	F 0
	▭	G 1	H 1	I 1

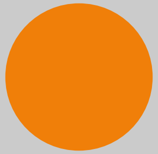
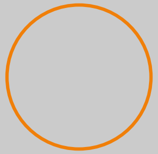
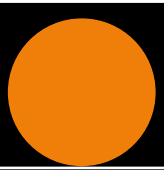
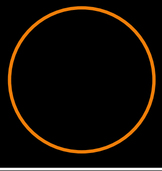
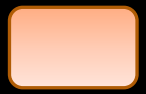
How do you compute $FP_{\square}$?

2. Generalizing the 2-by-2 contingency table

Predicted

Actual



		Correct Value		
				
Guessed Value		A 2	B 0	C 1
		D 1	E 2	F 0
		G 1	H 1	I 1

How do you compute $FP_{\square}$?

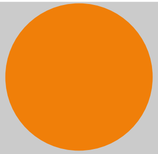
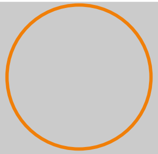
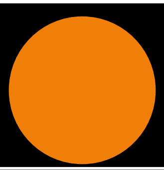
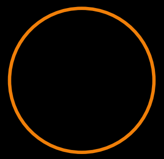
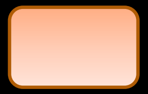
Generalizing the 2-by-2 contingency table

Q: Is this a good result?

		Correct Value		
		●	○	▭
Guessed Value	●	80	9	11
	○	7	86	7
	▭	2	8	9

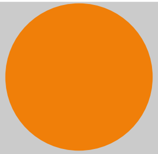
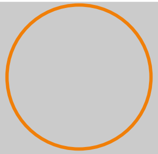
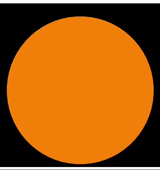
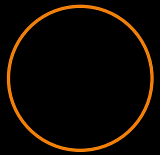
Generalizing the 2-by-2 contingency table

Q: Is this a good result?

		Correct Value		
				
Guessed Value		30	40	30
		25	30	50
		30	35	35

Generalizing the 2-by-2 contingency table

Q: Is this a good result?

		Correct Value		
				
Guessed Value		7	3	90
		4	8	88
		3	7	90