CS 360: Machine Learning

Prof. Sara Mathieson

Fall 2020

HAVERFORD
COLLEGE

Admin

- **Lab 6 due Wed**
  - Grace period until Thurs night
  - Help hours Sun/Mon/Tues

- Welcome prospective students!

Outline for October 23

- Evaluation Metrics
  - Confusion matrices revisited
  - ROC curves
  - Relationship to probabilistic methods
- Recap Random forests
- AdaBoost

Outline for October 23

- Evaluation Metrics
  - Confusion matrices revisited
  - ROC curves
  - Relationship to probabilistic methods
- Recap Random forests
- AdaBoost

For now: assume binary classification task

- Transactions that indicate credit card fraud
- Detecting which scans show tumors
- Prenatal test for Down's Syndrome
- Finding genes under natural selection
- Finding regions of the genome with high recombination rate ("hotspots")

For now: assume binary classification task

- Transactions that indicate credit card fraud
- Detecting which scans show tumors
- Prenatal test for Down's Syndrome
- Finding genes under natural selection
- Finding regions of the genome with high recombination rate ("hotspots")

In all these examples, we are trying to find unusual items ("needle in a haystack") -- we call these *positives*

Goals of Evaluation

- Think about what metrics are important for the problem at hand
- Compare different methods on the same problem
- Common set of tools that other researchers/users can understand

Precision and Recall

- Precision: of all the "flagged" examples, which ones are actually relevant (i.e. positive)?

- Recall: of all the relevant results, which ones did I actually return?

Modified from Ameet Soni

Precision and Recall

- Precision: of all the "flagged" examples, which ones are actually relevant (i.e. positive)?

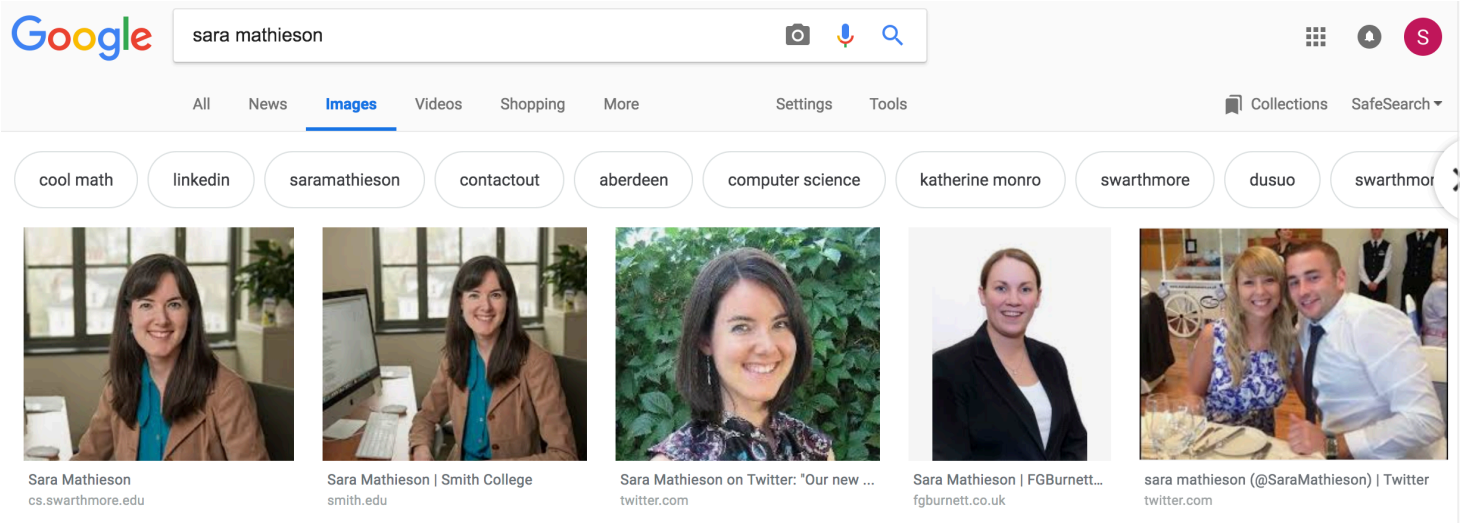
  (Purity)

- Recall: of all the relevant results, which ones did I actually return?

  (Completeness)

Modified from Ameet Soni

Precision and Recall

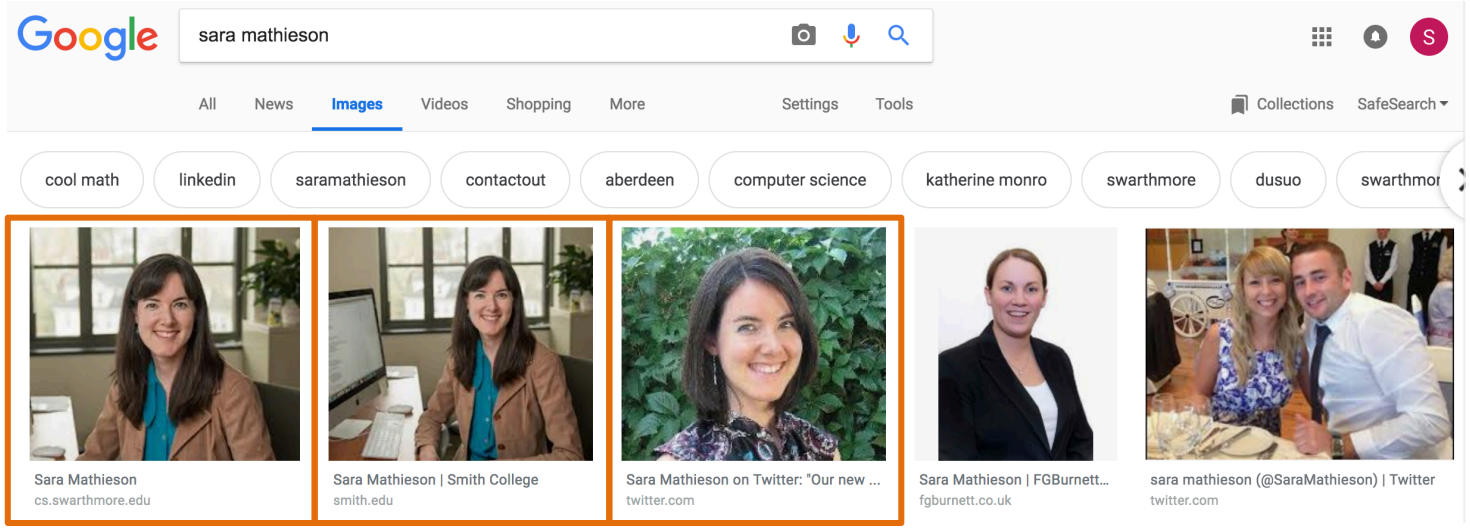


P=6 (number of images that are actually me)

- Precision?
- Recall?

Modified from Ameet Soni

Precision and Recall

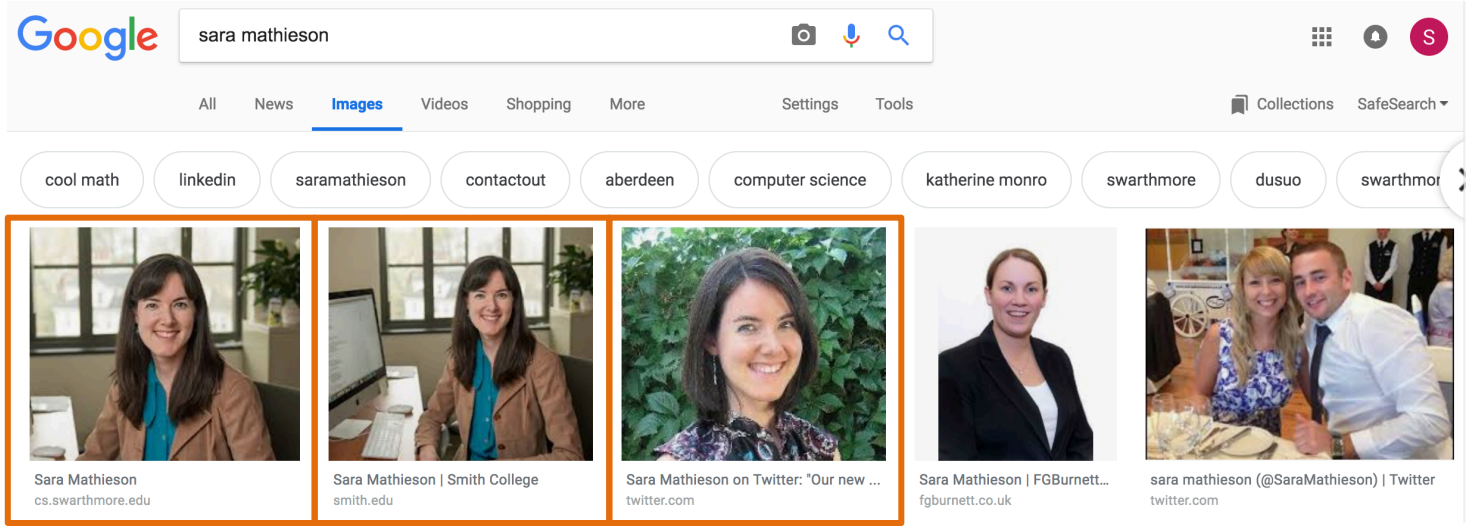


P=6 (number of images that are actually me)

- Precision = TP/(FP+TP) = 3/5
- Recall?

Modified from Ameet Soni

Precision and Recall

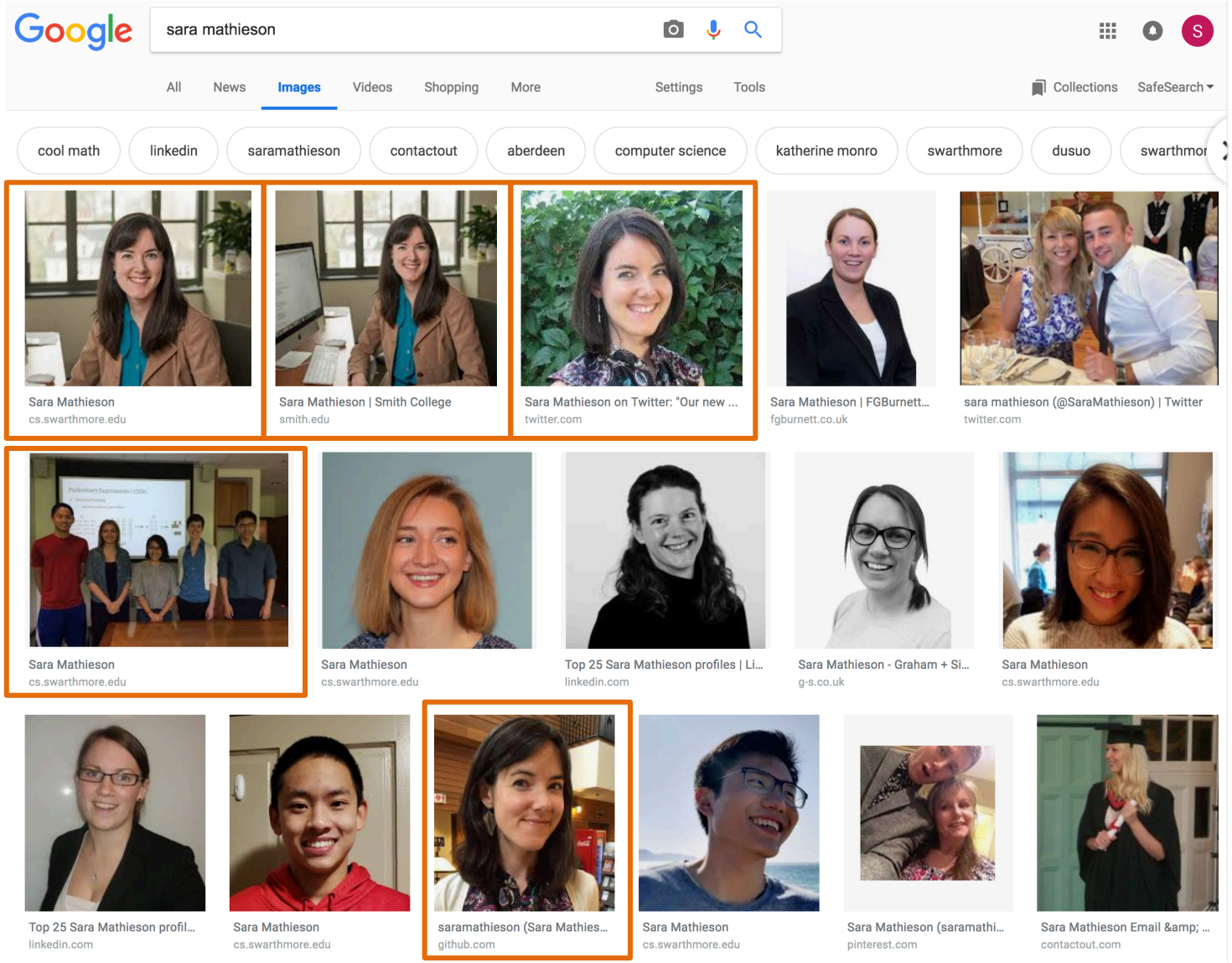


P=6 (number of images that are actually me)

- Precision = TP/(FP+TP) = 3/5
- Recall = TP/(FN+TP) = 3/6

Modified from Ameet Soni

Precision and Recall



P=6 (number of images that are actually me)

- Precision = 5/16
- Recall = 5/6

Modified from Ameet Soni

Revisit Confusion Matrices

		Predicted class	
		Negative	Positive
True class	Negative	True negative (TN)	False positive (FP)
	Positive	False negative (FN)	True positive (TP)

Modified from Jessica Wu

Revisit Confusion Matrices

True class \ Predicted class	Negative	Positive	
Negative	True negative (TN)	False positive (FP) "false alarm"	N (total number of true negatives)
Positive	False negative (FN) "miss"	True positive (TP)	P (total number of true positives)
	N* (what we said was negative)	P* (what we said was positive "flagged")	

Modified from Jessica Wu

Revisit Confusion Matrices

		Predicted class		
		Negative	Positive	
True class	Negative	True negative (TN) ✓	False positive (FP) "false alarm" ✗	N
	Positive	False negative (FN) "miss" ✗	True positive (TP) ✓	P
		N*	P*	

Modified from Jessica Wu

Revisit Confusion Matrices

True class \ Predicted class	Negative	Positive	
Negative	True negative (TN)	False positive (FP) "false alarm"	N
Positive	False negative (FN) "miss"	True positive (TP)	P
	N*	P*	

Error:

(FN+FP)/(TN+FP+FN+TP)

= (FN+FP)/(N+P)

Modified from Jessica Wu

Revisit Confusion Matrices

		Predicted class		
		Negative	Positive	
True class	Negative	True negative (TN)	False positive (FP) "false alarm"	N
	Positive	False negative (FN) "miss"	True positive (TP)	P
		N*	P*	

Accuracy = 1-Error:

(TN+TP)/(TN+FP+FN+TP)

= (TN+TP)/(N+P)

Modified from Jessica Wu

Revisit Confusion Matrices

True class \ Predicted class	Negative	Positive	
Negative	True negative (TN)	False positive (FP) "false alarm"	N
Positive	False negative (FN) "miss"	True positive (TP)	P
	N*	P*	

Precision:

TP/(FP+TP) = TP/P*

Modified from Jessica Wu

Revisit Confusion Matrices

		Predicted class		
		Negative	Positive	
True class	Negative	True negative (TN)	False positive (FP) "false alarm"	N
	Positive	False negative (FN) "miss"	True positive (TP)	P
		N*	P*	

Recall
(True Positive Rate):

TP/(FN+TP) = TP/P

Modified from Jessica Wu

Revisit Confusion Matrices

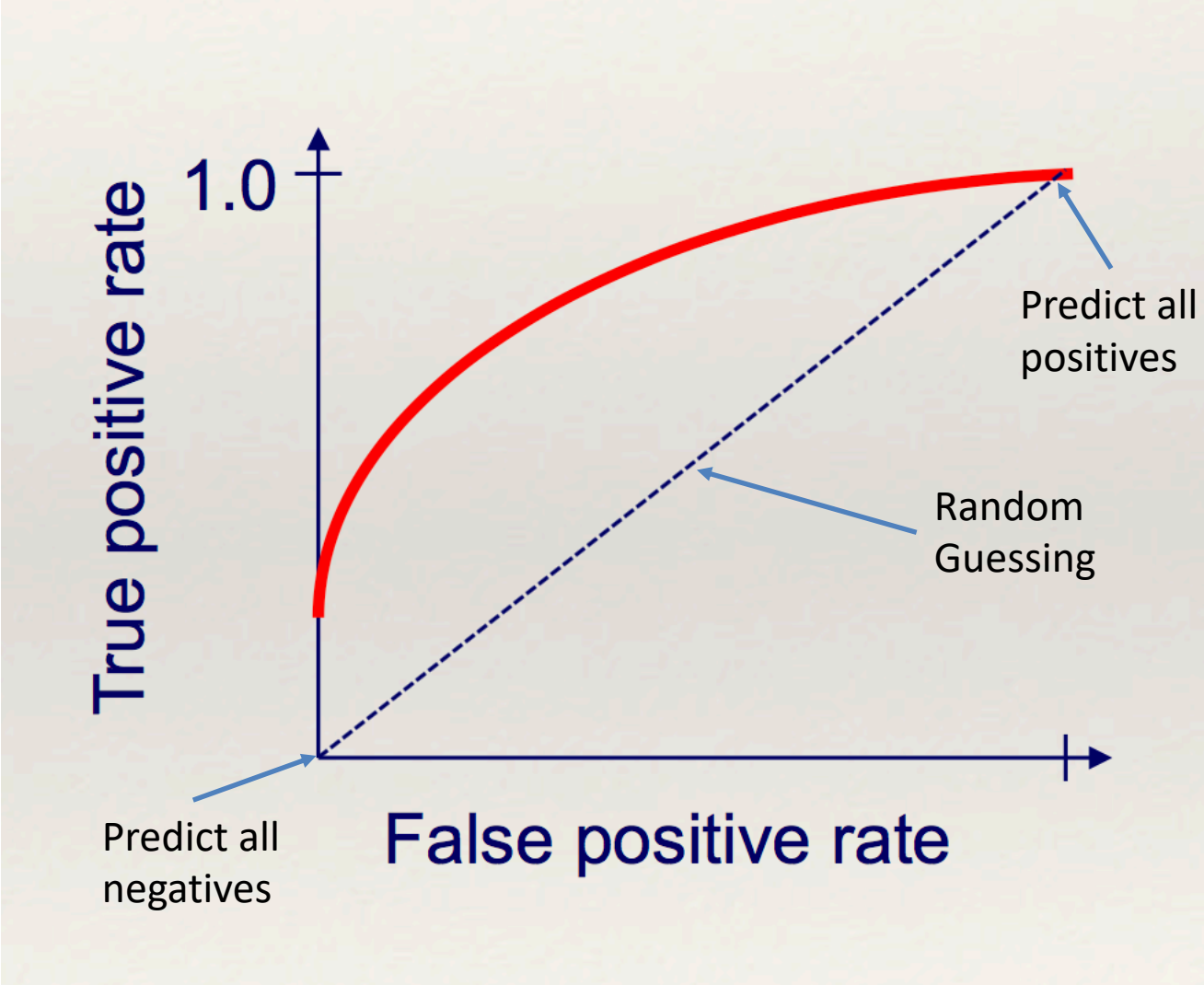
True class \ Predicted class	Negative	Positive	
Negative	True negative (TN)	False positive (FP) "false alarm"	N
Positive	False negative (FN) "miss"	True positive (TP)	P
	N*	P*	

False Positive Rate:

FP/(TN+FP) = FP/N

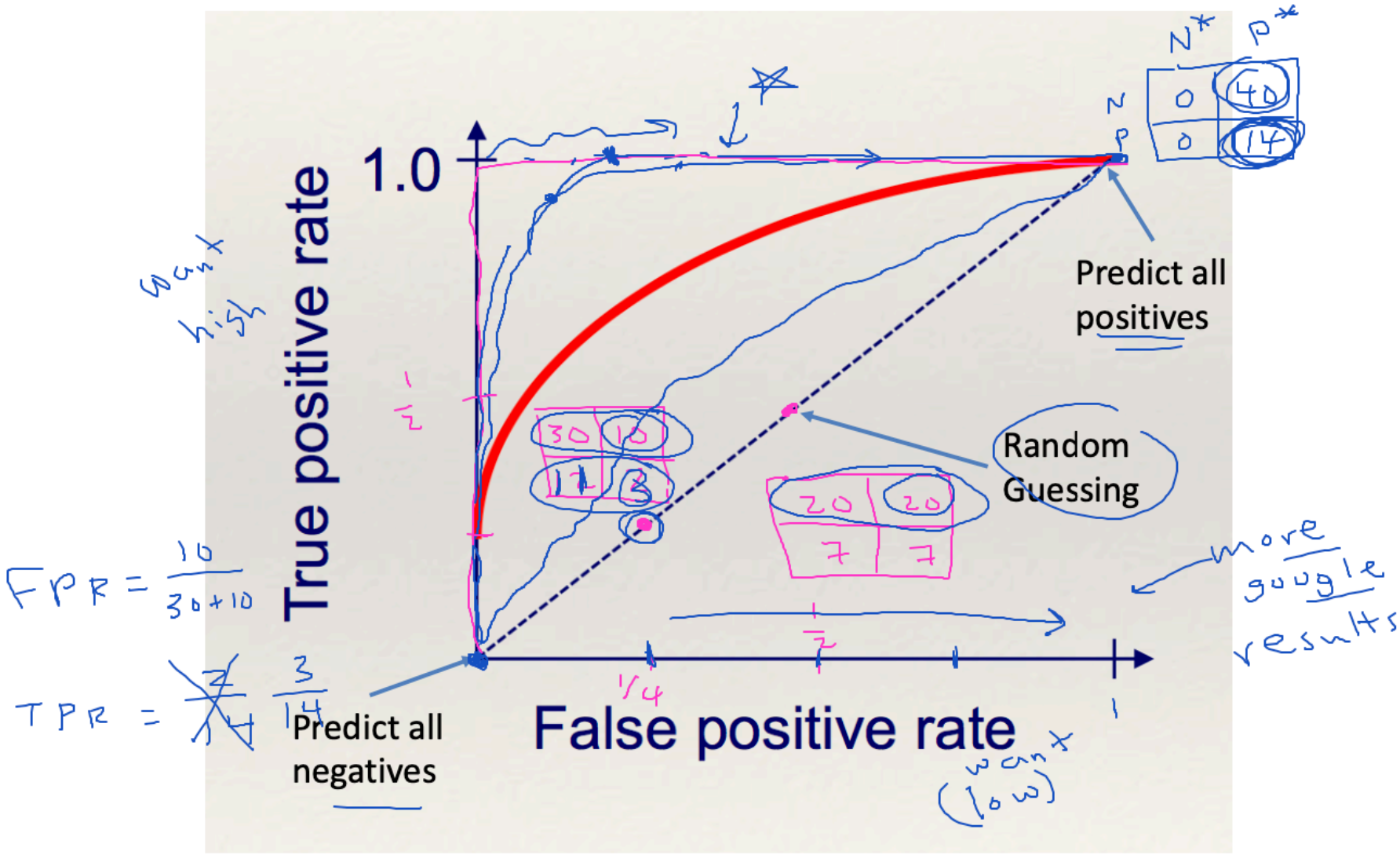
Modified from Jessica Wu

ROC curve (Receiver Operating Characteristic)



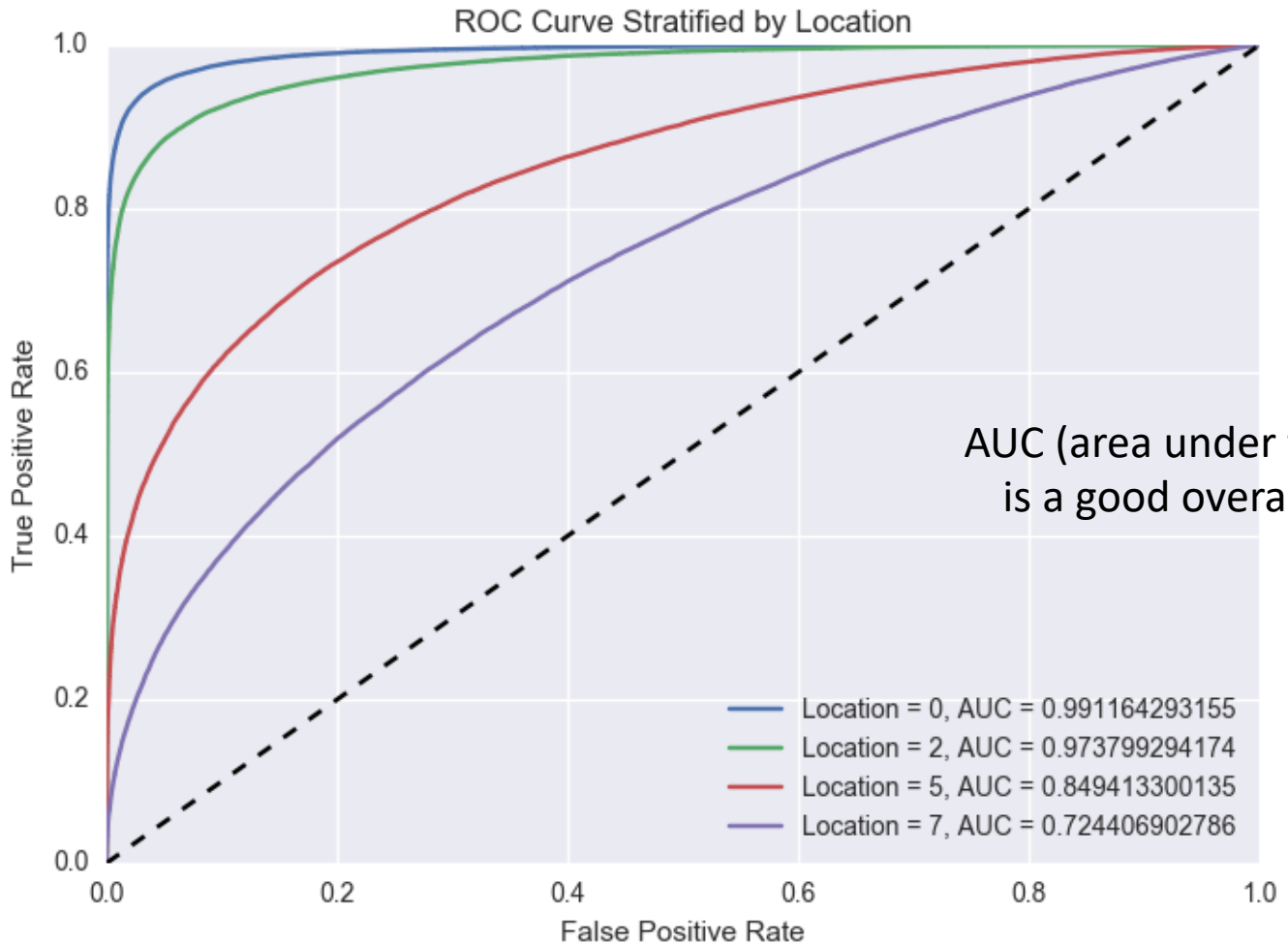
Modified from Ameet Soni

ROC curve (Receiver Operating Characteristic)



Modified from Ameet Soni

ROC curve example: comparing methods



AUC (area under the curve)
is a good overall metric

Example of a ROC curve from my research
Chan, Perrone, Spence, Jenkins, Mathieson, Song

How to get a ROC curve for probabilistic methods?

- Usually we use 0.5 as a threshold for binary classification

- Vary the threshold! (i.e. choose 0.25)

  - $P(y=1 \mid x) > 0.25$ => classify as 1 (positive)
  - $P(y=1 \mid x) <= 0.25$ => classify as 0 (negative)

How to get a ROC curve for probabilistic methods?

- Usually we use 0.5 as a threshold for binary classification

- Vary the threshold! (i.e. choose 0.25)

  - P(y=1 | x) > 0.25 => classify as 1 (positive)
  - P(y=1 | x) <= 0.25 => classify as 0 (negative)

$p(y=1|x) > 0.9$ => few positives

Handout 12

(1)

pred

true

77	3
13	7

→ N = 80

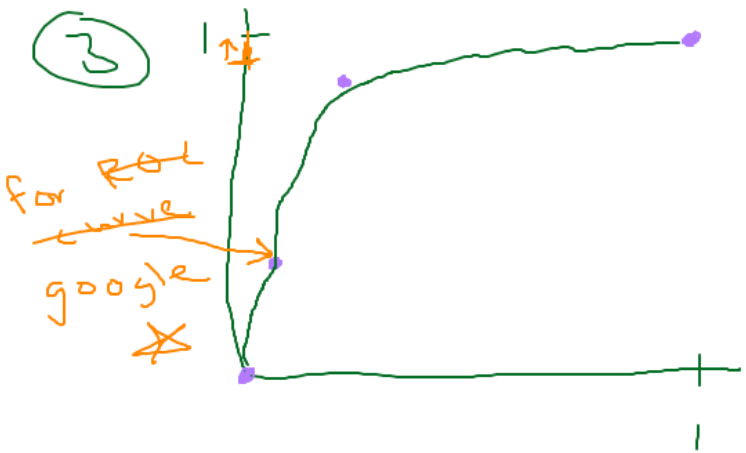
→ P = 20

(2) precision $\frac{7}{3+7} = 0.7$

recall = TPR

$= \frac{7}{20} = 0.35$

(3)



$FPR = \frac{3}{80} = 0.04$

(4)

68	12
2	18

$FPR = \frac{12}{80} = 0.15$

$TPR = \frac{18}{20} = 0.9$

Outline for October 23

- Evaluation Metrics
  - Confusion matrices revisited
  - ROC curves
  - Relationship to probabilistic methods
- Recap Random forests
- AdaBoost

Random Forests

- Idea: choose a different **subset of features** for every classifier *t*
- Typically use *decision stumps* (depth 1)

- Goal: decorrelate models

- In practice: choose **sqrt(p)** features
  - Without replacement for each model
  - Every model: data points and features chosen independently

Outline for October 23

- Evaluation Metrics
  - Confusion matrices revisited
  - ROC curves
  - Relationship to probabilistic methods
- Recap Random forests
- AdaBoost

Boosting Idea

TRAIN

- Start with all example equally weighted
- For T iterations:
  - Learn weak classifier using weighted examples
  - Change example weights

Boosting Idea

TRAIN

- Start with all example equally weighted
- For T iterations:
  - Learn weak classifier using weighted examples
  - Change example weights

TEST

- Get predictions from all weak classifiers
- Vote based on how well each classifier did during training

AdaBoost (adaptive boosting)

start: $\boxed{w_i^{(1)} = \frac{1}{n}}$ for all training examples

for $t = 1 \ldots T$ ← # classifiers

(a) fit classifier to weighted examples

$\Rightarrow \boxed{h^{(t)}(\vec{x})}$

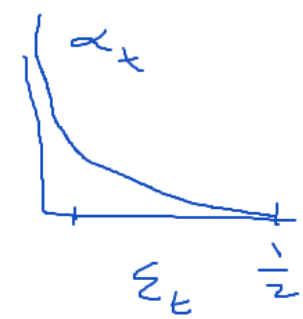
(b) weighted class. error

want low → $\varepsilon_t = \sum_{i=1}^{n} w_i^{(t)} \mathbb{1}\left(\underbrace{y_i}_{\text{true}} \neq \underbrace{h^{(t)}(\vec{x}_i)}_{\text{pred}} \right)$

error

(c) compute score ← want high

$$\alpha_t = \frac{1}{2} \log\left(\frac{1 - \varepsilon_t}{\varepsilon_t} \right)$$



AdaBoost (adaptive boosting)

model

test

$$\underbrace{h}(x) = \text{sign}\left(\sum_{t=1}^{T} \underbrace{\alpha_t}_{\text{model score}} \overbrace{h^{(t)}(x)}^{\{-1,1\}}\right)$$

$$\alpha_t = \frac{1}{2} \underbrace{\log\left(\frac{1-\varepsilon_t}{\varepsilon_t}\right)}$$