Analysis of AI Systems

CS 640
Margrit Betke

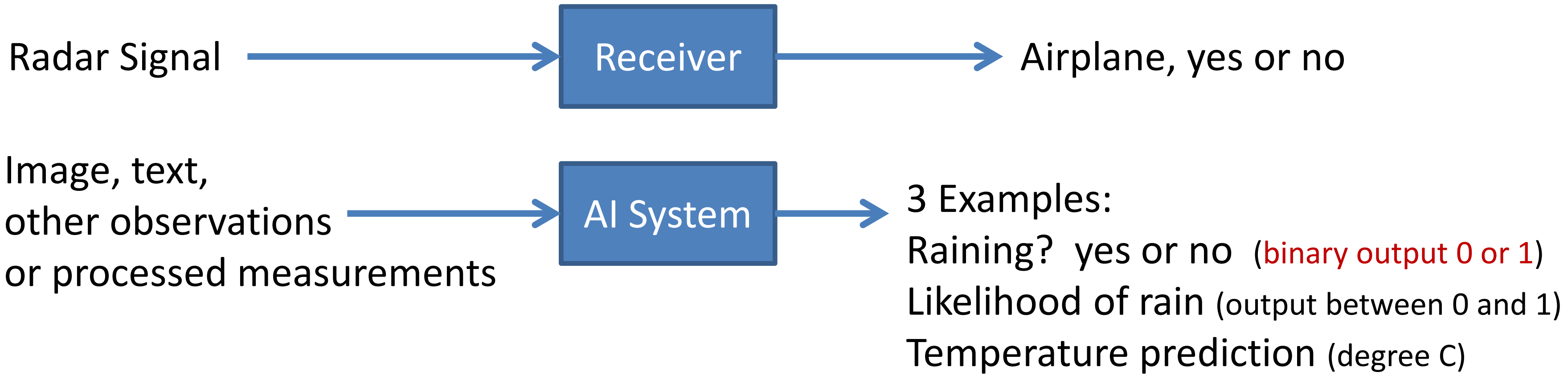
Lecture 2

September 7, 2023

ROC Analysis

ROC = receiver operating characteristics   (historic name from radar signal processing)

ROC Analysis = Method to organize, visualize, and evaluate results of an AI system



ROC Analysis

Image, text,
other observations
or processed measurements → AI System ↓

<u>4 Examples:</u>	<u>Predictor Type:</u>
1) Raining? yes or no (binary output 0 or 1)	Binary classifier
2) Likelihood of rain (output between 0 and 1)	
Threshold on likelihood is typically used to make binary decision	
3) Rain? Sun? Snow?	Multiclass Classifier
"One-hot" Output: (1,0,0) => rain	
or Likelihood score: (0.1, 0.7, 0.2) => sun	
4) Temperature prediction (degree C)	Regressor

Evaluating a Regressor

"Truth" =
Ground truth =
Gold standard =
Actual value =

$y_1,\ldots,y_n$

AI System output =
Hypothesis =
Predicted value=

$\hat{y}_1,\ldots,\hat{y}_n$

Evaluating a Regressor, e.g., Temperature Predictor

"Truth" =
Ground truth =
Gold standard =
Actual value =

$y_1,\ldots,y_n$

Compare measured temperature with Predicted temperature:

Error: $y_i - \hat{y}_i$

AI System output =
Hypothesis =
Predicted value=

$\hat{y}_1,\ldots,\hat{y}_n$

e.g., 80F – 78F = 2F error

Evaluating a Regressor, e.g., Temperature Predictor

"Truth" =
Ground truth =
Gold standard =
Actual value =

$y_1, \ldots, y_n$

AI System output =
Hypothesis =
Predicted value=

$\hat{y}_1, \ldots, \hat{y}_n$

Compare measured temperature with Predicted temperature:

$$\text{Error: } y_i - \hat{y}_i$$

e.g., 80F – 85F = -5F error

Need error measure that handles positive and negative differences!

Evaluating a Regressor, e.g., Temperature Predictor

"Truth" =
Ground truth =
Gold standard =
Actual value =

$y_1,\ldots,y_n$

Compare measured temperature with Predicted temperature:

Error: $| y_i - \hat{y}_i |$

AI System output =
Hypothesis =
Predicted value=

$\hat{y}_1,\ldots,\hat{y}_n$

Absolute Error?

Evaluating a Regressor, e.g., Temperature Predictor

"Truth" =
Ground truth =
Gold standard =
Actual value =

$y_1,\ldots,y_n$

AI System output =
Hypothesis =
Predicted value=

$\hat{y}_1,\ldots,\hat{y}_n$

Compare measured temperature with Predicted temperature:

$$\text{Error: } (y_i - \hat{y}_i)^2$$

Squared error is preferred. Why?

Evaluating a Regressor over full dataset:

"Truth" =
Ground truth =
Gold standard =
Actual value =

$y_1,\ldots,y_n$

Mean Squared Error:

$$\text{MSE} = 1/n \sum_{i=1}^{n} (y_i - \hat{y}_i)^2$$

Or

Root Mean Squared Error:

AI System output =
Hypothesis =
Predicted value=

$\hat{y}_1,\ldots,\hat{y}_n$

$$\text{RMSE} = \sqrt{1/n \sum_{i=1}^{n} (y_i - \hat{y}_i)^2}$$

Confusion Matrix for Binary Output Case

"Truth” =
Ground truth =
Gold standard =
Actual class

AI System output =
Hypothesis =
Predicted class

	1	0
1	True Positive (TP)	False Positive (FP)
0	False Negative (FN)	True Negative (TN)

Confusion Matrix for Binary Output Case

Example with 20 samples

"Truth" =
Ground truth =
Gold standard =
Actual class

AI System output =
Hypothesis =
Predicted class

	1	0
1	True Positive (TP): 6	False Positive (FP): 4
0	False Negative (FN): 2	True Negative (TN): 8

Confusion Matrix for Binary Output Case

"Truth" =
Ground truth =
Gold standard =
Actual class

AI System output =
Hypothesis =
Predicted class

	1	0
1	True Positive (TP): 6	False Positive (FP): 4
0	False Negative (FN): 2	True Negative (TN): 8

1st step of analyzing the confusion matrix:
Check that sum of matrix entries = number of samples used to test AI system

Confusion Matrix for Binary Output Case

"Truth" =
Ground truth =
Gold standard =
Actual class

AI System output =
Hypothesis =
Predicted class

	1	0
1	True Positive (TP): 6	False Positive (FP): 4
0	False Negative (FN): 2	True Negative (TN): 8

Good System?

Confusion Matrix for Binary Output Case

"Truth" =
Ground truth =
Gold standard =
Actual class

AI System output =
Hypothesis =
Predicted class

	1	0
1	True Positive (TP): 6	False Positive (FP): 4
0	False Negative (FN): 2	True Negative (TN): 8

Good System? We want high values in diagonal of matrix.

TP+TN=6+8=14

Confusion Matrix for Binary Output Case

"Truth" =
Ground truth =
Gold standard =
Actual class

AI System output =
Hypothesis =
Predicted class

	1	0
1	True Positive (TP): 6	False Positive (FP): 4
0	False Negative (FN): 2	True Negative (TN): 8

2nd step of analyzing the confusion matrix:
Compute sum of diagonal entries and compare that with total number of samples

Confusion Matrix for Binary Output Case

TP+TN=6+8=14
Total number of samples = 20

14 versus 20: Is this a good system?

"Truth" =
Ground truth =
Gold standard =
Actual class

AI System output =
Hypothesis =
Predicted class

	1	0
1	True Positive (TP): 6	False Positive (FP): 4
0	False Negative (FN): 2	True Negative (TN): 8

2^{nd} step of analyzing the confusion matrix:
Compute sum of diagonal entries and compare that with total number of samples

Confusion Matrix for Binary Output Case

TP+TN=6+8=14

Total number of samples = 20

Accuracy of AI System:

14/20 = 0.7

"Truth" = Ground truth = Gold standard = Actual class

AI System output = Hypothesis = Predicted class

	1	0
1	**True Positive (TP): 6**	**False Positive (FP): 4**
0	**False Negative (FN): 2**	**True Negative (TN): 8**

2nd step of analyzing the confusion matrix:

Compute sum of diagonal entries and compare that with total number of samples

Confusion Matrix for Binary Output Case

"Truth" =
Ground truth =
Gold standard =
Actual class

AI System output =
Hypothesis =
Predicted class

	1	0
1	True Positive (TP): 6	False Positive (FP): 4
0	False Negative (FN): 2	True Negative (TN): 8

Positive samples = TP+FN = 8

Negative samples = FP+FN = 12

How sensitive is the classifier in finding the positives?

true positive rate = tp = TP/(TP+FN) = 6/8 = ¾ = ***recall*** = ***sensitivity***

"Truth" = Ground truth = Gold standard = Actual class

AI System output = Hypothesis = Predicted class

	1	0
1	**True Positive (TP): 6**	**False Positive (FP): 4**
0	**False Negative (FN): 2**	**True Negative (TN): 8**

Positive samples = P= TP+FN = 8

false positive rate = fp = FP/(FP+TN) = 4/12 = 1/3

"Truth" = Ground truth = Gold standard = Actual class

AI System output = Hypothesis = Predicted class

	1	0
1	True Positive (TP): 6	False Positive (FP): 4
0	False Negative (FN): 2	True Negative (TN): 8

Negative samples = FP+TN = 12

true positive rate = **tp** =
TP/(TP+FN) = 6/8 = ¾
= ***recall*** = ***sensitivity***

false positive rate = **fp** =
FP/(FP+TN) = 4/12 = 1/3

"Truth" =
Ground truth =
Gold standard =
Actual class

AI System output =
Hypothesis =
Predicted class

	1	0
1	**True Positive (TP): 6**	**False Positive (FP): 4**
0	**False Negative (FN): 2**	**True Negative (TN): 8**

Positive samples =
TP+FN =
8

Negative samples =
FP+TN =
12

How specific is the classifier in finding the negatives?

Instead of *fp*, we sometimes focus on

1 – *fp* = *<u>specificity</u>*
TN/(FP+TN) = 8/12 = 2/3

"Truth" =
Ground truth =
Gold standard =
Actual class

AI System output =
Hypothesis =
Predicted class

	1	0
1	**True Positive (TP): 6**	**False Positive (FP): 4**
0	**False Negative (FN): 2**	**True Negative (TN): 8**

Negative samples =
FP+TN =
12

How precise is the classifier in finding the positives?

true positive rate **= tp = TP/(TP+FN) = 6/8 = ¾ = *recall* = *sensitivity***

precision **= TP/(TP+FP) = 6/10 = 3/5**

"Truth" = Ground truth = Gold standard = Actual class

AI System output = Hypothesis = Predicted class

	1	0
1	**True Positive (TP): 6**	**False Positive (FP): 4**
0	**False Negative (FN): 2**	**True Negative (TN): 8**

Positive hypotheses = **TP**+**FP** = 10

Positive samples = TP+FN = 8

F1 Score

true positive rate **= tp = TP/(TP+FN) = 6/8 = ¾ = *recall* = *sensitivity***

precision **= TP/(TP+FP) = 6/10 = 3/5**

F1 score = 2 recall x precision/(recall + precision)
= 2 x 3/4 x 3/5 / (3/4+3/5) = 2/3

AI System output = Hypothesis = Predicted class

	1	0
1	**True Positive (TP): 6**	**False Positive (FP): 4**
0	**False Negative (FN): 2**	**True Negative (TN): 8**

Positive hypotheses = **TP+FP** = 10

Positive samples = TP+FN = 8

F1 Score

true positive rate **= tp =**
TP/(TP+FN) = 6/8 = ¾ = 0.75
= ***recall*** **=** ***sensitivity***

precision **=**
TP/(TP+FP) = 6/10 = 3/5 = 0.6

F1 score = 2 recall x precision/(recall + precision)
= 2 x 3/4 x 3/5 / (3/4+3/5) = 2/3 = **0.667**

AI System output =
Hypothesis =
Predicted class

	1	0
1	**True Positive (TP): 6**	**False Positive (FP): 4**
0	**False Negative (FN): 2**	**True Negative (TN): 8**

Balanced Accuracy

true positive rate **= tp =**
TP/(TP+FN) = 6/8 = ¾ = 0.75
= *recall* = *sensitivity*

Balanced Accuracy = (((TP/(TP+FN)+(TN/(TN+FP))) / 2 =
(sensitivity + specificity)/2 =
(3/4 + 2/3)/2 = (0.75 + 0.67)/2 = 2/3 = **0.708**

1 – *fp* = *specificity*
TN/(FP+TN) = 8/12 = 0.67

AI System output =
Hypothesis =
Predicted class

	1	0
1	**True Positive (TP): 6**	**False Positive (FP): 4**
0	**False Negative (FN): 2**	**True Negative (TN): 8**

Accuracy vs. F1 Score vs. Balanced Accuracy

Accuracy = (TP+TN)/everything = **0.700**

F1 Score = 2 recall x precision/(recall + precision) = **0.667**

Balanced Accuracy = (sensitivity + specificity)/2 = **0.708**

	1	0
1	**True Positive (TP): 6**	**False Positive (FP): 4**
0	**False Negative (FN): 2**	**True Negative (TN): 8**

Accuracy vs. F1 Score vs. Balanced Accuracy

Accuracy = (TP+TN)/everything = **0.999**

F1 Score = 2 recall x precision/(recall + precision) = **0.667**

Balanced Accuracy = (sensitivity + specificity)/2 = **0.833**

	1	0
1	**True Positive (TP): 6**	**False Positive (FP): 4**
0	**False Negative (FN): 2**	**True Negative (TN): 8000**

Accuracy vs. F1 Score vs. Balanced Accuracy

Accuracy = (TP+TN)/everything = **0.999**

F1 Score = 2 recall x precision/(recall + precision) = **0.999**

Balanced Accuracy = (sensitivity + specificity)/2 = **0.833**

	1	0
1	**True Positive (TP):6000**	**False Positive (FP): 4**
0	**False Negative (FN): 2**	**True Negative (TN): 8**

Terms to remember:

ROC

Classifier

Predictor

Likelihood

Ground truth, gold standard
Hypothesis

Accuracy, Balanced Accuracy, F1 score

False positive rate & False negative rate

Recall & Precision

Sensitivity & Specificity

Building an ROC curve for an AI System: One classifier at time

TP+TN=6+8=14

Total number of samples = 20

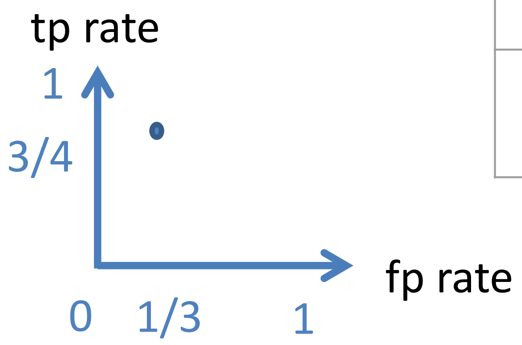
Accuracy of AI System:

14/20 = 0.7

false positive rate = 1/3

true positive rate = 3/4

ROC curve has 1 point:



"Truth" =
Ground truth =
Gold standard =
Actual class

	1	0
1	**True Positive (TP): 6**	**False Positive (FP): 4**
0	**False Negative (FN): 2**	**True Negative (TN): 8**

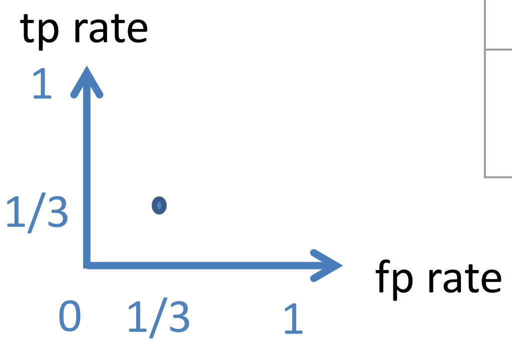
Good Classifier?

"Truth" =
Ground truth =
Gold standard =
Actual class

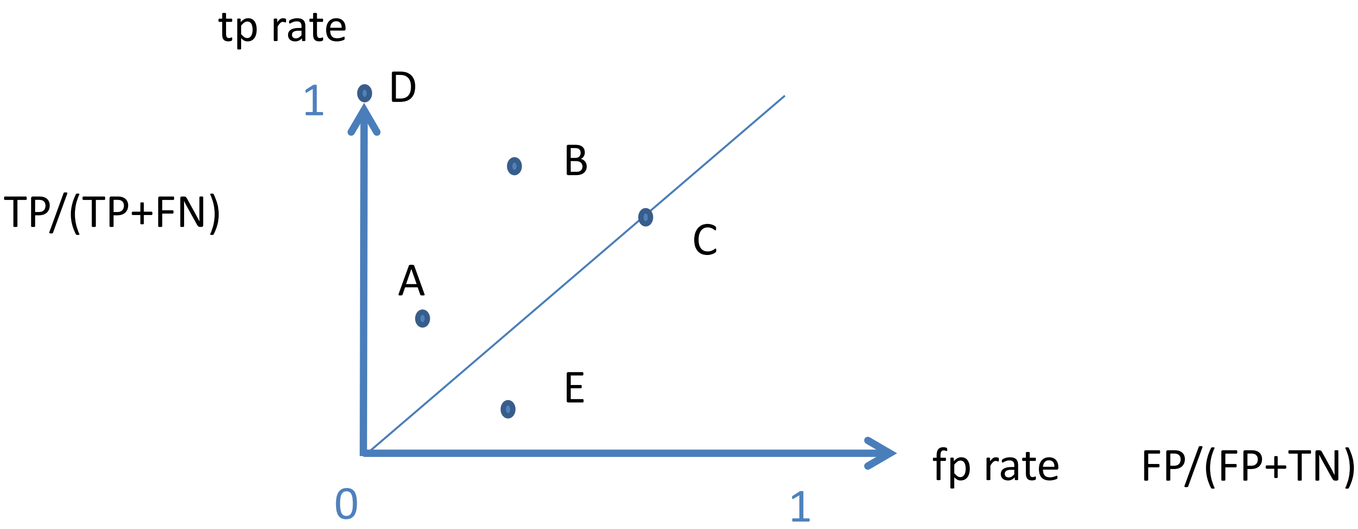
	1	0
1	True Positive (TP): 4	False Positive (FP): 4
0	False Negative (FN): 8	True Negative (TN): 8

false positive rate = 1/3
true positive rate = 1/3

ROC curve has 1 point:



Comparing Classifiers



Classifier A:

Classifier B:

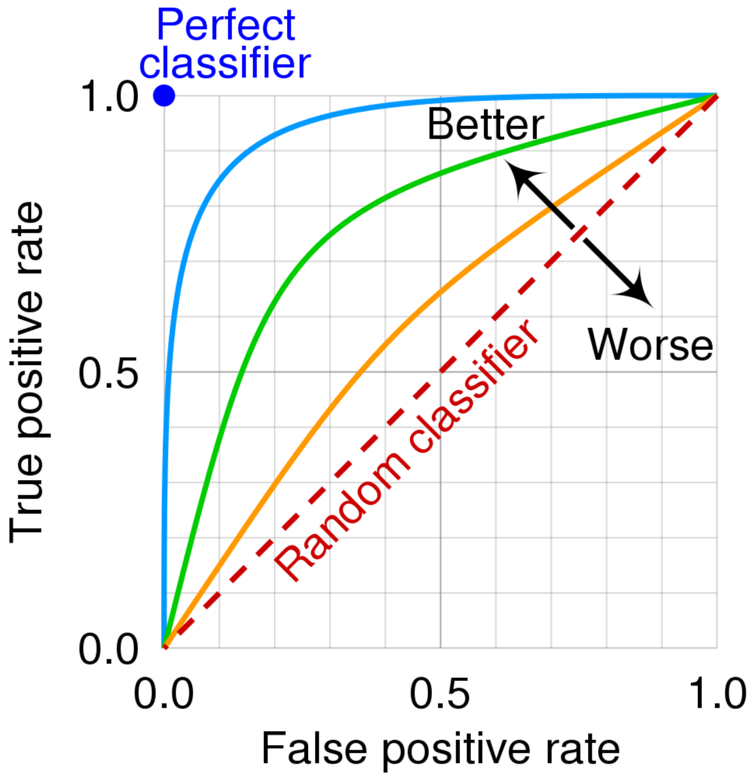
Classifier C:

Classifier D:

Classifier E:

See paper by Fawcett

ROC Curves



Each colored line shows the behavior of a binary classifier when a parameter is changed.

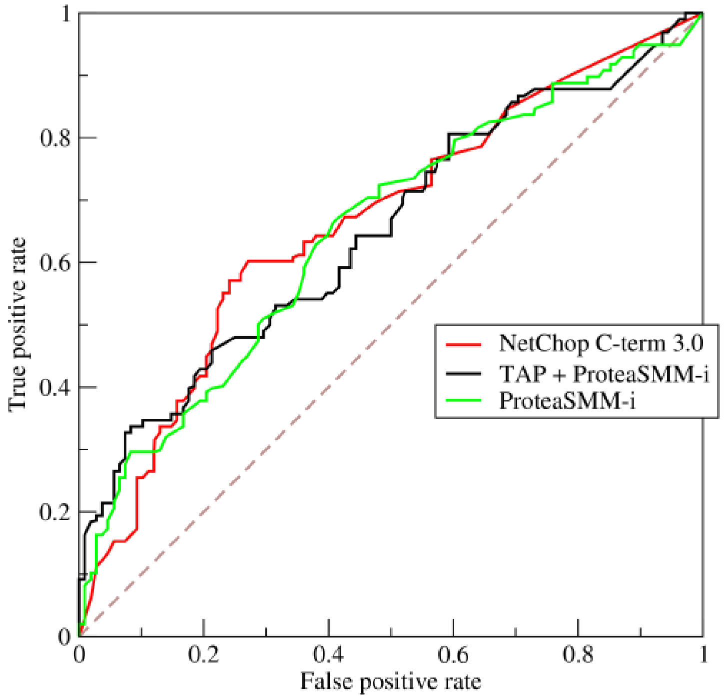
Example for the rain prediction classifier:
The parameter could be the threshold T on the likelihood of its prediction for rain:

Likelihood	> T	Predict "rain"
	<= T	Predict "no rain"

Drawn by CMG Lee based on https://commons.wikimedia.org/wiki/File:Roc_curve.svg

ROC Curve: Classifier

Real example: Three predictors of peptide cleaving



On the Quest to Interpret Web Image Content: Salient Object Subitizing

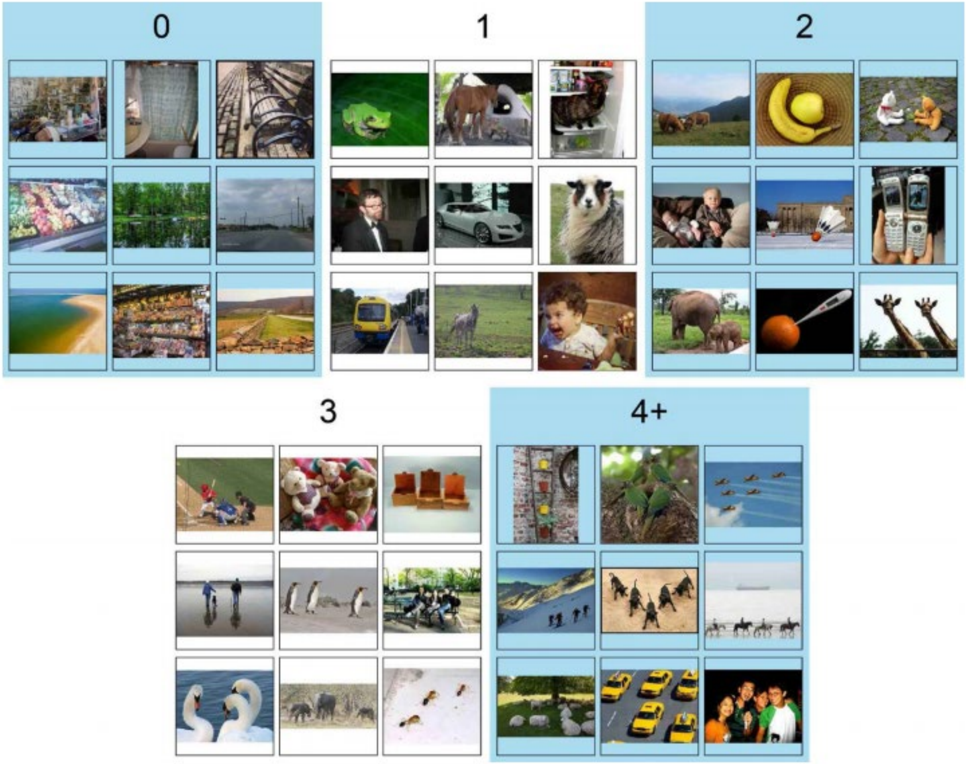
Jianming Zhang, Shugao Ma,
Mehrnoosh Sameki, Stan Sclaroff,
Margrit Betke, et al.,

CVPR 2015
IJCV 2017

Salient Object Subitizing

Task:
Predict the existence and number of salient objects in a scene

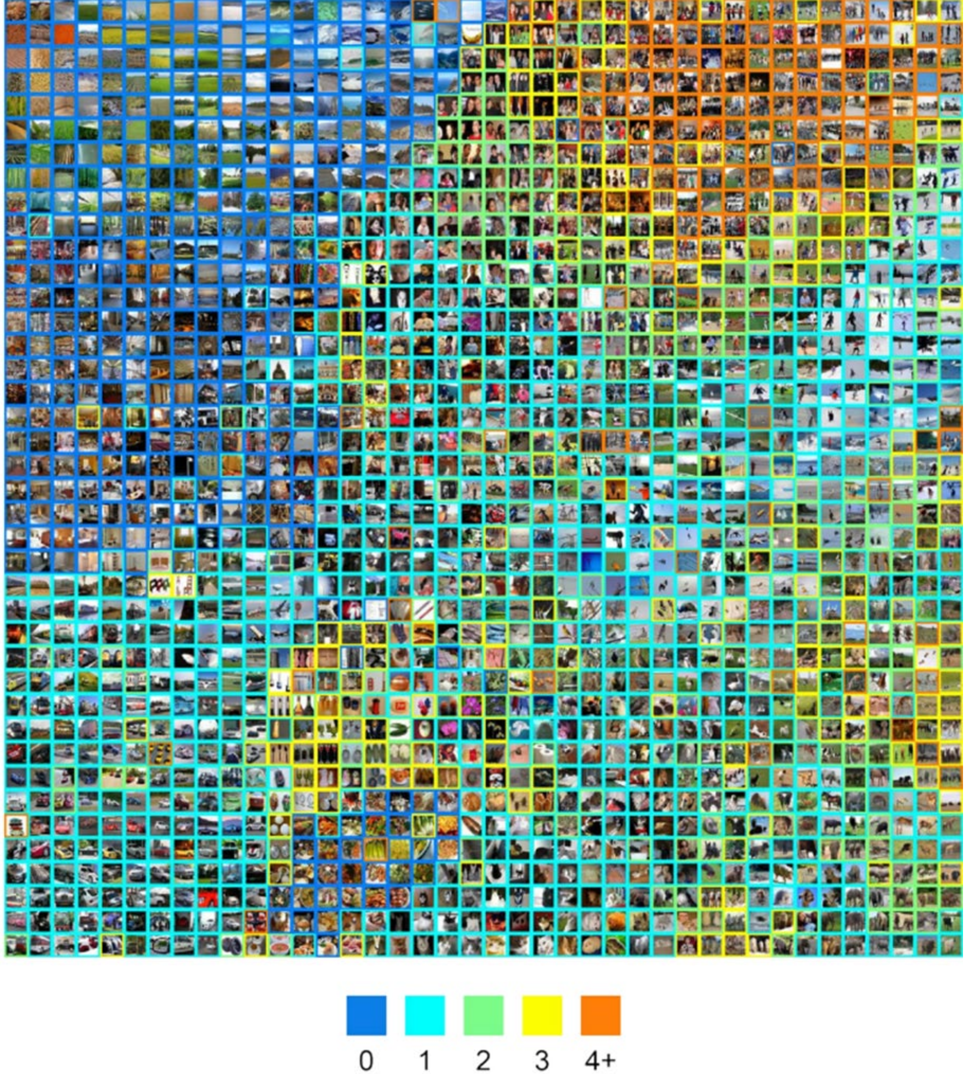
Solution:
GoogLeNet CNN called SOS



Zhang et al.

~ 69% accurate

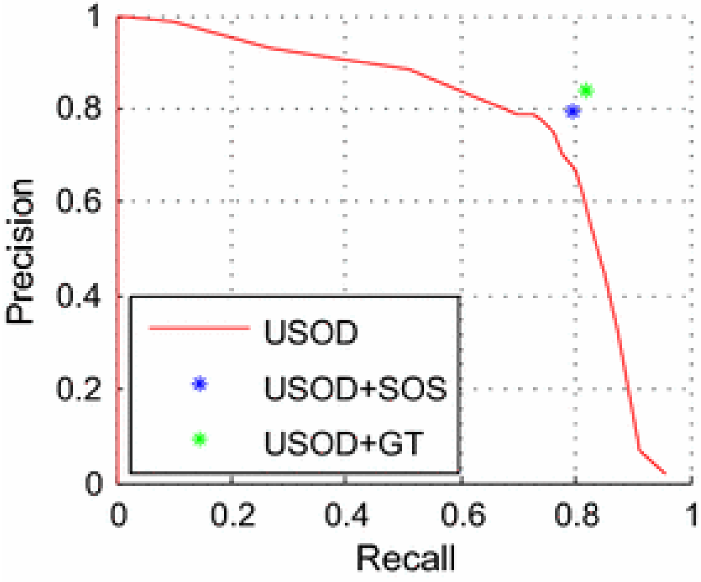
0 or 1 object:
>90% accurate



Comparing Classifiers

Some researchers prefer to draw precision/recall curves instead of tp/fp curves:

Precision
=TP/(TP+FP)



Recall = tp = TP/(TP+FN)

Plot from one of my research papers
Zhang et al, IJCV 2017:

Classifier: Salient object subitizing (SOS)
= predicts the number (1, 2, 3, and 4+) of
salient objects in an image
USOD stands for “unconstrained object detection method”
(parameter: number of detection windows)
GT = ground truth

Confusion Matrix for Multiple Classes

"Truth" = Ground truth = Gold standard = Actual class

AI System Output = Hypothesis = Predicted class

	Class 1	Class 2	Class 3	Class 4
Class 1	100%	15%	10%	7%
Class 2	0%	80%	10%	3%
Class 3	0%	3%	80%	70%
Class 4	0%	2%	0%	20%

Confusion Matrix for Multiple Classes

"Truth" = Ground truth = Gold standard = Actual class

AI System Output = Hypothesis = Predicted class

	Class 1	Class 2	Class 3	Class 4
Class 1	100%	15%	10%	7%
Class 2	0%	80%	10%	3%
Class 3	0%	3%	80%	70%
Class 4	0%	2%	0%	20%

Confusion Matrix for Multiple Classes

Note: Rows and columns of a confusion matrix may be reversed
Reporting only percentages and not actual number is usually NOT a good practice.

Example of a multi-class confusion matrix in one of my papers (Zhang et al, IJCV 2017):

Each row corresponds to a ground-truth category label. The percentage reported is the average proportion of images of the category A (row number) labeled as category B (column number). For over 90% images, predicted labels are consistent with the ground-truth labels.

	0	1	2	3	4+
0	90% (179)	5% (9)	2% (3)	1% (2)	3% (6)
1	1% (2)	96% (191)	3% (5)	1% (1)	1% (1)
2	0	3% (6)	95% (189)	3% (5)	0
3	0	1% (1)	3% (5)	96% (191)	1% (2)
4+	13% (26)	3% (6)	4% (8)	2% (3)	78% (156)