CS340 Machine learning
ROC curves

Performance measures for binary classifiers

Confusion matrix, contingency table

		y	
$\hat{y}$	1	0	
1	TP	FP	$\hat{P}$
0	FN	TN	$\hat{N}$
	P	N	

precision = positive predictive value (PPV) = TP / P-hat

Sensitivity = recall =
True pos rate = hit rate
= TP / P = 1-FNR

False pos rate = false acceptance =
= type I error rate = FP / N = 1-spec

False neg rate = false rejection =
type II error rate = FN / P = 1-TPR

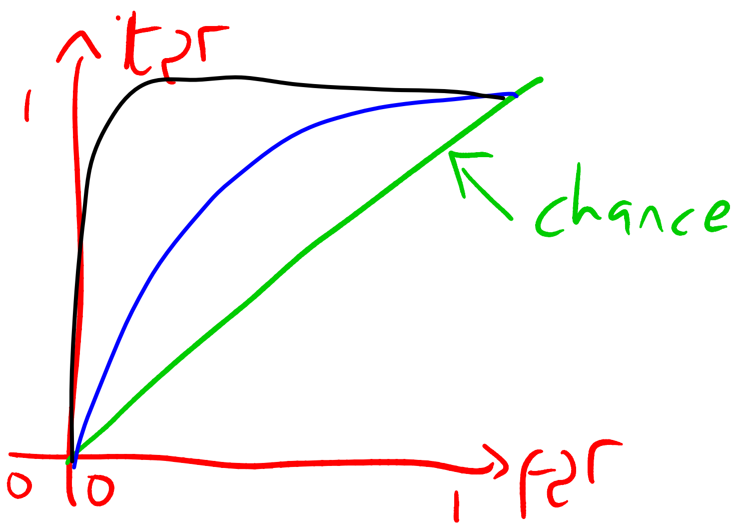
Specificity = TN / N = 1-FPR

Performance depends on threshold

- Declare x_n to be a positive if $p(y=1|x_n)>\theta$, otherwise declare it to be negative (y=0)

$$\hat{y}_n = 1 \iff p(y=1|x_n) > \theta$$

- Number of TPs and FPs depends on threshold θ. As we change θ, we get different (TPR, FPR) points.



$$TPR = p(\hat{y}=1|y=1)$$

$$FPR = p(\hat{y}=1|y=0)$$

Example

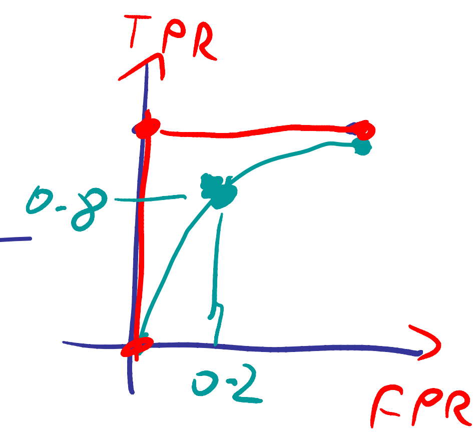
i	y_i	$p(y_i = 1\|x_i)$	$\hat{y}_i(\theta = 0)$	$\hat{y}_i(\theta = 0.5)$	$\hat{y}_i(\theta = 1)$
1	1	0.9	1	1	0
2	1	0.8	1	1	0
3	1	0.7	1	1	0
4	1	0.6	1	1	0
5	1	0.5	1	1	0
6	0	0.4	1	0	0
7	0	0.3	1	0	0
8	0	0.2	1	0	0
9	0	0.1	1	0	0

TPR = 5/5 = 1
FPR = 4/4 = 1

TPR = 5/5 = 1
FPR = 0/4 = 0

TPR = 0/5 = 0
FPR = 0/4 = 0

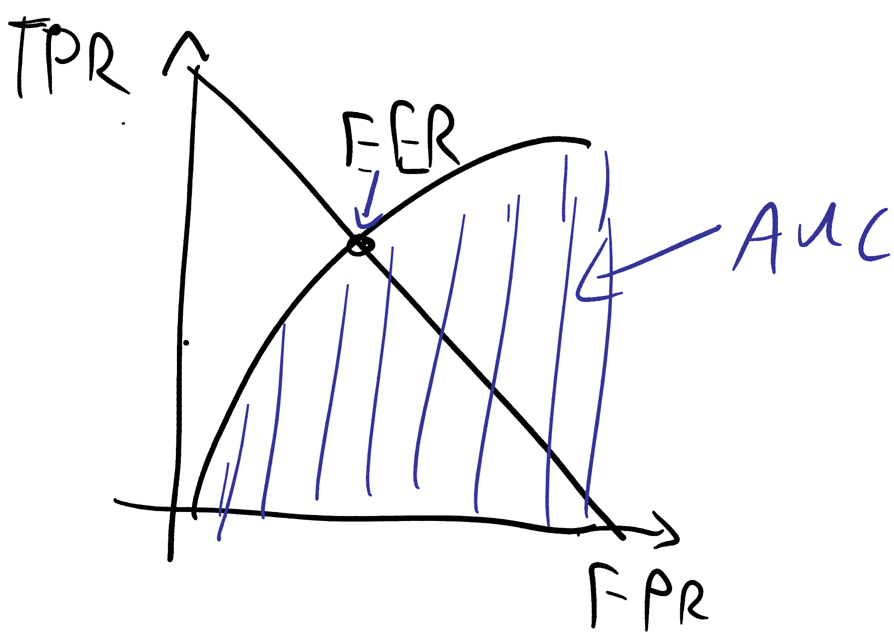
i	y_i	$p(y_i = 1\|x_i)$	$\hat{y}_i(\theta = 0)$	$\hat{y}_i(\theta = 0.5)$	$\hat{y}_i(\theta = 1)$
1	1	0.9	1	1	0
2	1	0.8	1	1	0
3	1	0.7	1	1	0
4	1	0.6	1	1	0
5	1	**0.2**	1	**0**	0
6	0	**0.6**	1	**1**	0
7	0	0.3	1	0	0
8	0	0.2	1	0	0
9	0	0.1	1	0	0



TPR = 4/5 = 0.8
FPR = 1/4 = 0.25

Performance measures

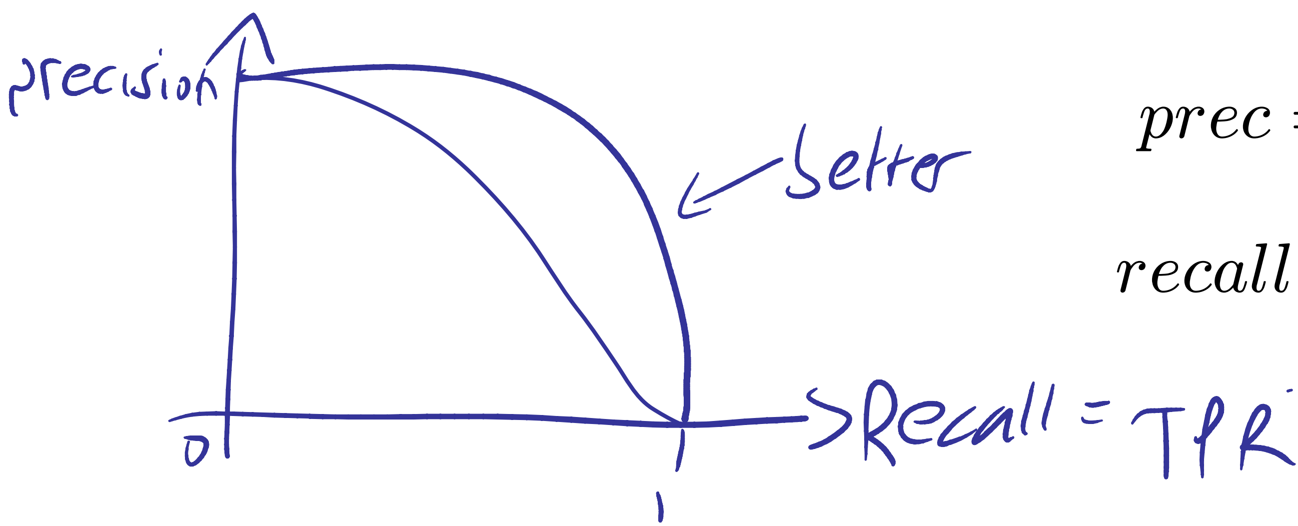
- EER- Equal error rate/ cross over error rate (false pos rate = false neg rate), smaller is better
- AUC - Area under curve, larger is better
- Accuracy = (TP+TN)/(P+N)



Precision-recall curves

- Useful when notion of "negative" (and hence FPR) is not well defined, or too many negatives (rare event detection)
- Recall = of those that exist, how many did you find?
- Precision = of those that you found, how many correct?
- F-score is harmonic mean

$$F = \frac{2}{1/P + 1/R} = \frac{2PR}{R+P}$$



$$prec = p(y=1|\hat{y}=1)$$

$$recall = p(\hat{y}=1|y=1)$$

Word of caution

- Consider binary classifiers A, B, C

y	A	.	B	.	C	.
$\hat{y}$	1	0	1	0	1	0
1	0.9	0.1	0.8	0	0.78	0
0	0	0	0.1	0.1	0.12	0.1

- Clearly A is useless, since it always predicts label 1, regardless of the input. Also, B is slightly better than C (less probability mass wasted on the off-diagonal entries). Yet here are the performance metrics.

Metric	A	B	C
Accuracy	0.9	0.9	0.88
Precision	0.9	1.0	1.0
Recall	1.0	0.888	0.8667
F-score	0.947	0.941	0.9286

Mutual information is a better measure

The MI between estimated and true label is

$$I(\hat{Y}, Y) = \sum_{\hat{y}=0}^{1} \sum_{y=0}^{1} p(\hat{y}, y) \log \frac{p(\hat{y}, y)}{p(\hat{y})p(y)}$$

This gives the intuitively correct rankings B>C>A

Metric	A	B	C
Accuracy	0.9	0.9	0.88
Precision	0.9	1.0	1.0
Recall	1.0	0.888	0.8667
F-score	0.947	0.941	0.9286
Mutual information	0	0.1865	0.1735