Evaluating Classifiers and Annotating Data

Kenneth (Kenny) Joseph

University at Buffalo
Department of Computer Science and Engineering
School of Engineering and Applied Sciences

3/15/22

Announcements

- Quiz 6 is "out"
- Midterm is **Thursday**
  - In class
  - One page handwritten notes, front and back
  - **Nothing else** (except pen/pencil)
- Two quick review things
- Questions?

Evaluating Classification Models

- How should we evaluate (part 1)?
- What is the best we can do?
- What is the worst we can do?
- Class Imbalances
- How should we evaluate (part 2)?
- Dealing w/ Class Imbalance through Modeling

UB has created a predictive algorithm to determine who should be admitted to the CSE MS program.

The algorithm takes the ~~ACT~~ GRE score and School Ranking as features, and past decisions on admissions as the outcome

The algorithm is used to **admit or reject students** starting next year

Back to regression

- How would we evaluate this with regression, i.e. what would our evaluation metric be?

$$\frac{1}{N}\sum(y - h(x))^2$$

What values can y take on? What about h(x)?

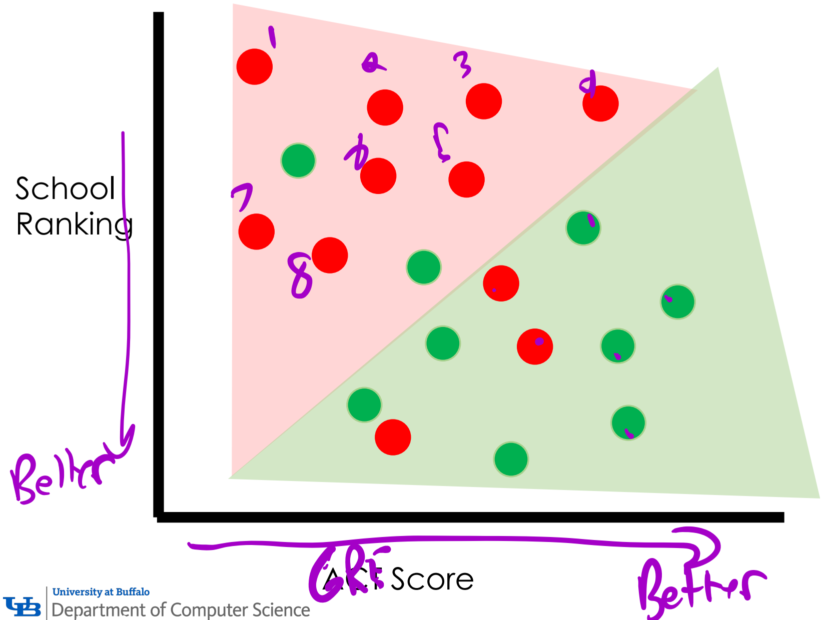
+1 -1 +1 -1

0

$$\sum(+1 - +1)^2 \qquad \sum(1 - -1)^2 = 4$$

0

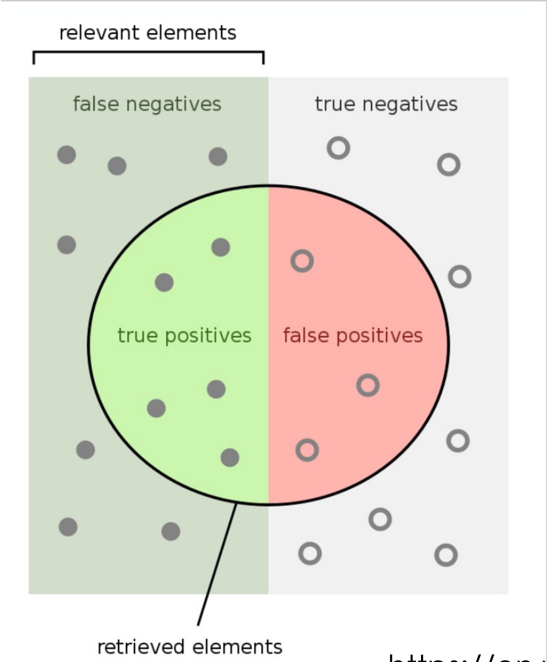
Evaluating classification models



Our guess:

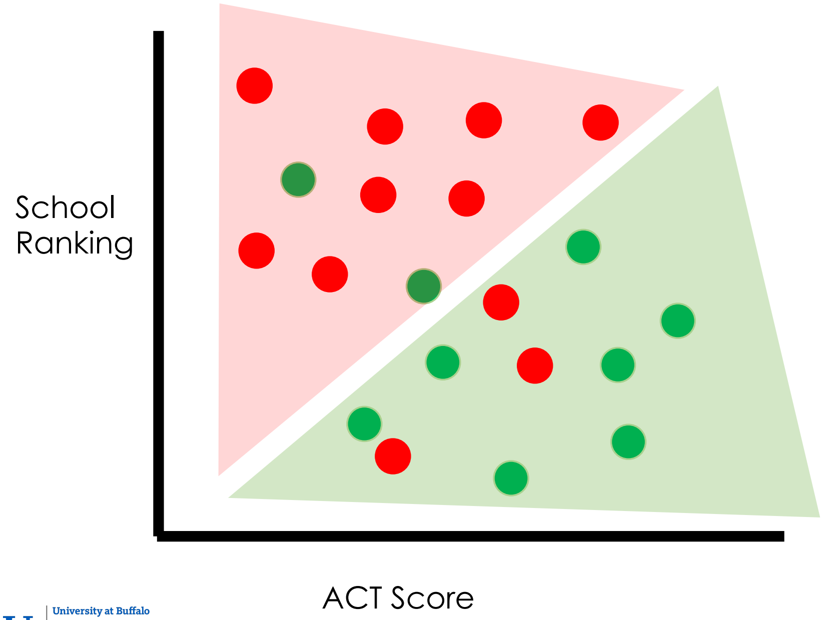
"Truth"		
	8	3
	2	7

The Confusion Matrix



https://en.wikipedia.org/wiki/Precision_and_recall

Accuracy – how many did we get correct?



Our guess:

"Truth"		
	8	3
	2	**7**

Accuracy =
(8 + 7) / (8 + 7 + 2 + 3)
= .75

What is the best we can do?

The Bayes optimal classifier

$y^* = h_{Best}(x) = \underset{x}{\operatorname{argmax}}\ \underbrace{P(y|\mathbf{x})}$ ← what if we know this

$\left[\begin{array}{l} p(+1|\mathbf{x}) = .8 \\ p(-1|\mathbf{x}) = .2 \end{array}\right.$ ← wrong 20% of the time!

$E_{opt} = 1 - P(h_{Best}(x)|x)$

Discussion follows: https://www.cs.cornell.edu/courses/cs4780/2018fa/lectures/lecturenote02_kNN.html

What is the worst we can do?

constant classifier / average label / majority

Random Guessing : 50%

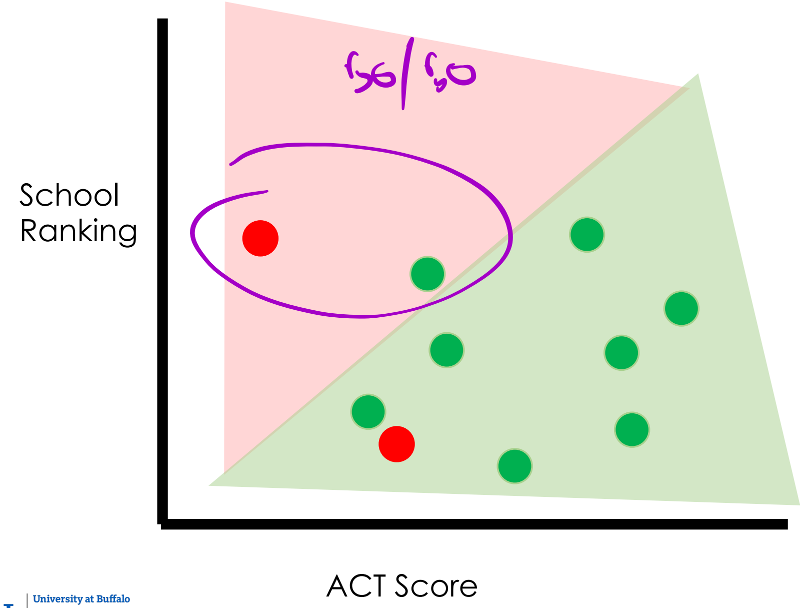
80% if P(+1) = .8

class imbalance

Always compare to the simple baseline for your model

Discussion follows: https://www.cs.cornell.edu/courses/cs4780/2018fa/lectures/lecturenote02_kNN.html

The problem with class imbalance



Our guess:

"Truth"

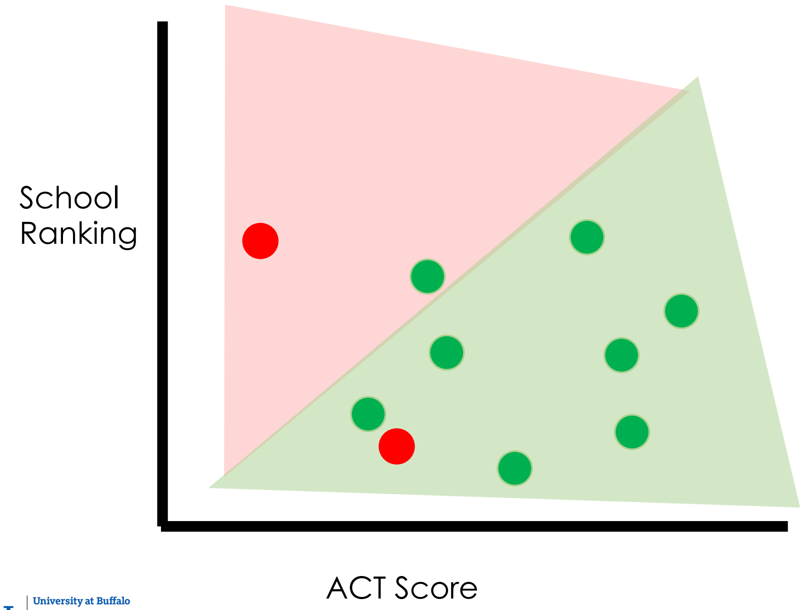
● (red)	1	1
● (green)	1	7

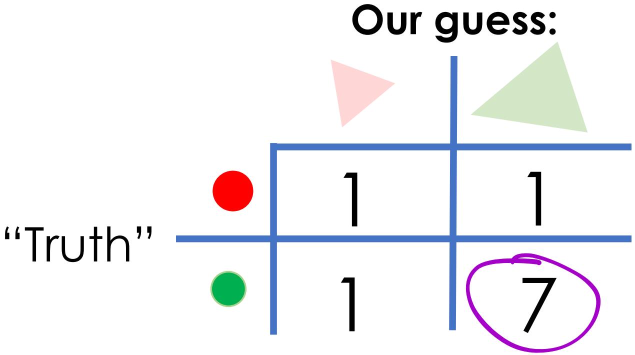
What is our accuracy?

(1+7)/((1+7+1+1))

80%

The problem with class imbalance



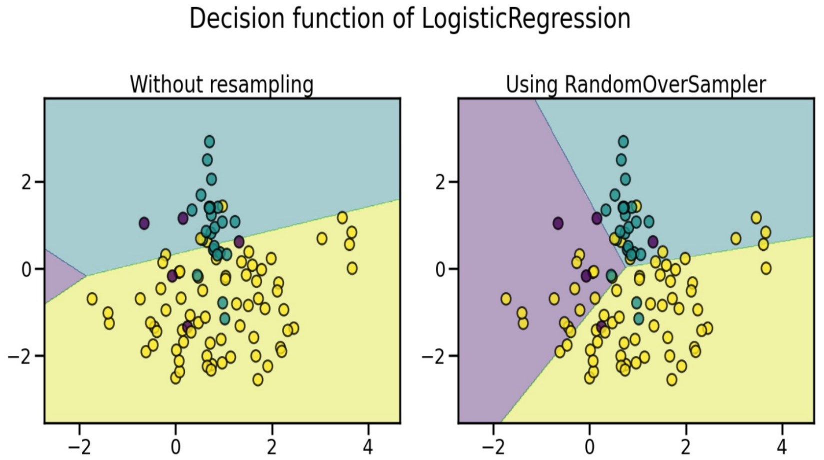


Is this really a good classifier? Not really

How does a majority classifier do?

70%

Aside – dealing with class imbalance

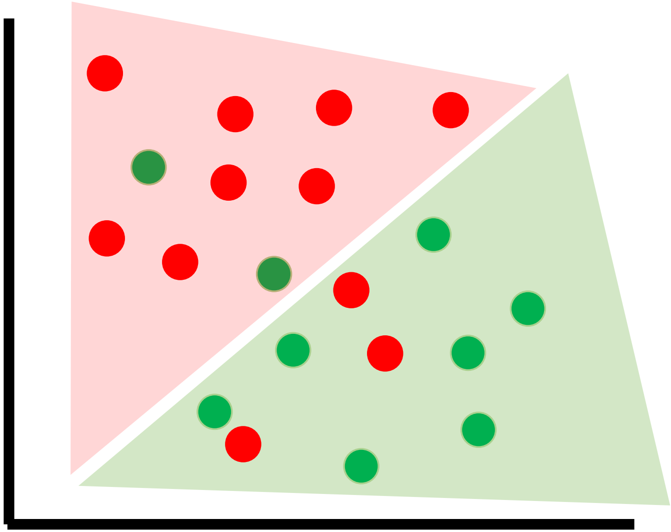


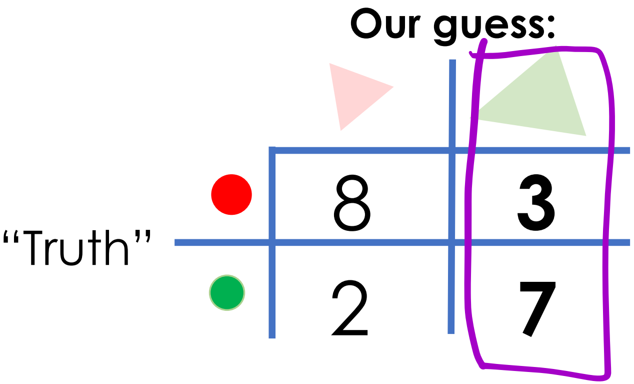
As a result, the majority class does not take over the other classes during the training process.

https://imbalanced-learn.org/stable/over_sampling.html

Will cover, along with a few other things, in a "practical issues" lecture at some point after the break

Precision - Of + guesses, how many actually +s?



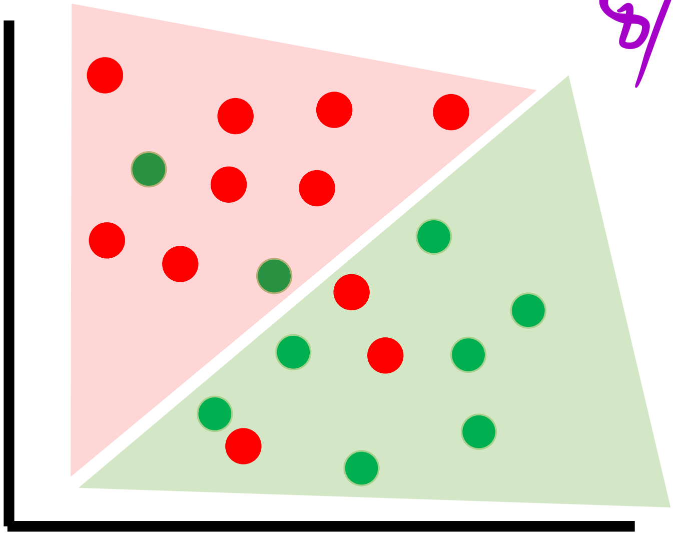


Precision =
$\underbrace{7}$ / $\underbrace{(7 + 3)}$ = .7

Recall - Of actual +, how many do we guess?

Recall of - class:

$8/(8+3)$

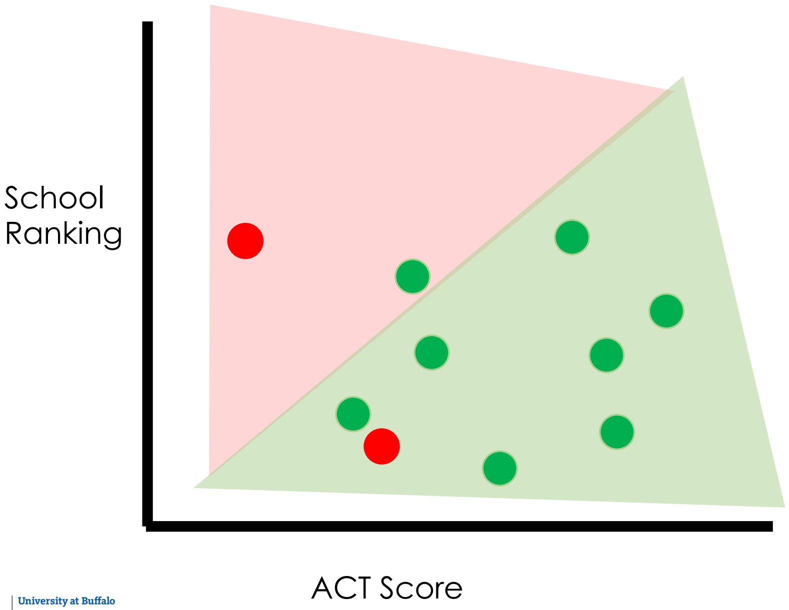


Our guess:

"Truth"		
	8	3
	2	**7**

Recall =
7 / (7 + 2) = .78

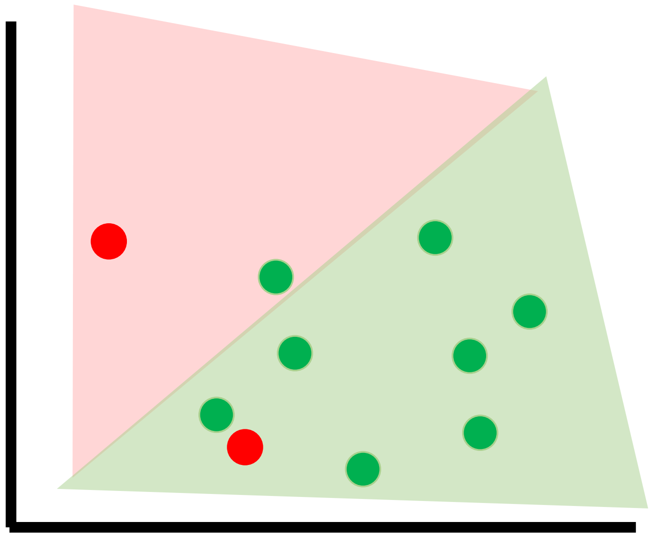
To compute precision and recall, you have to pick a class!



Recall - : 50%

Precision - : 50%

To compute precision and recall, you have to pick a class!

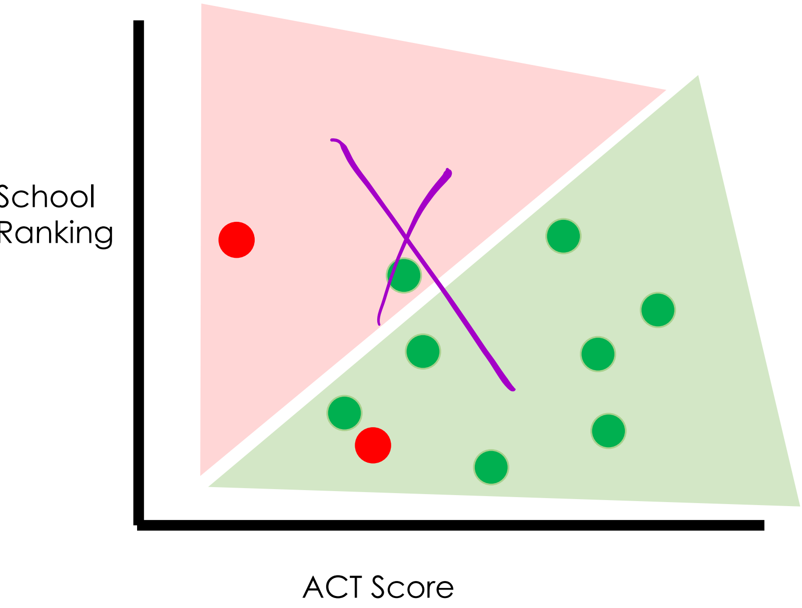


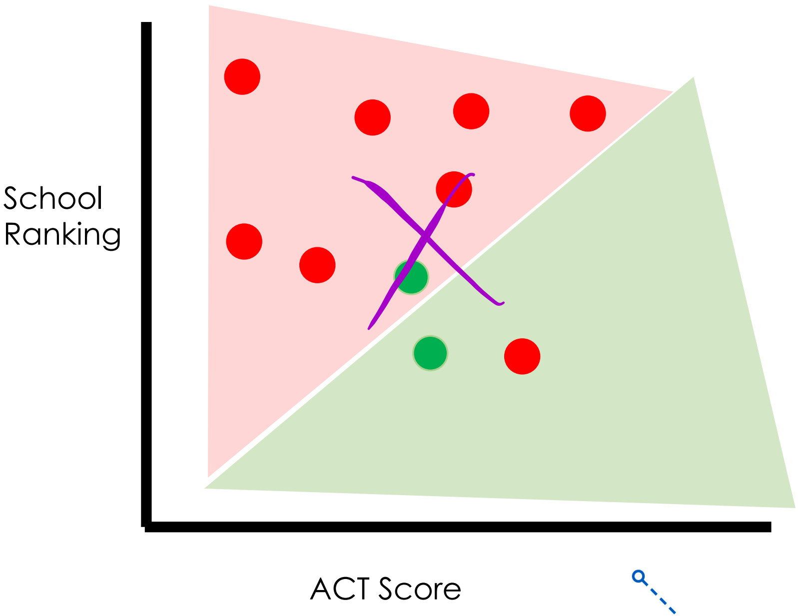
If you were applying to UB,

Recall +

Precision +

Which would you prefer?





There are many other metrics

Other metrics can be included in a confusion matrix, each of them having their significance and use.

Sources: [20][21][22][23][24][25][26][27] view · talk · edit

		Predicted condition			
	Total population $= P + N$	Positive (PP)	Negative (PN)	Informedness, bookmaker informedness (BM) $= TPR + TNR - 1$	Prevalence threshold (PT) $= \frac{\sqrt{TPR \times FPR} - FPR}{TPR - FPR}$
Actual condition	Positive (P)	**True positive (TP), hit**	**False negative (FN), type II error, miss, underestimation**	True positive rate (TPR), recall, sensitivity (SEN), probability of detection, hit rate, power $= \frac{TP}{P} = 1 - FNR$	False negative rate (FNR), miss rate $= \frac{FN}{P} = 1 - TPR$
Actual condition	Negative (N)	**False positive (FP), type I error, false alarm, overestimation**	**True negative (TN), correct rejection**	False positive rate (FPR), probability of false alarm, fall-out $= \frac{FP}{N} = 1 - TNR$	True negative rate (TNR), specificity (SPC), selectivity $= \frac{TN}{N} = 1 - FPR$
	Prevalence $= \frac{P}{P + N}$	Positive predictive value (PPV), precision $= \frac{TP}{PP} = 1 - FDR$	False omission rate (FOR) $= \frac{FN}{PN} = 1 - NPV$	Positive likelihood ratio (LR+) $= \frac{TPR}{FPR}$	Negative likelihood ratio (LR–) $= \frac{FNR}{TNR}$
	Accuracy (ACC) $= \frac{TP + TN}{P + N}$	False discovery rate (FDR) $= \frac{FP}{PP} = 1 - PPV$	Negative predictive value (NPV) $= \frac{TN}{PN} = 1 - FOR$	Markedness (MK), deltaP (Δp) $= PPV + NPV - 1$	Diagnostic odds ratio (DOR) $= \frac{LR+}{LR-}$
	Balanced accuracy (BA) $= \frac{TPR + TNR}{2}$	F_1 score $= \frac{2PPV \times TPR}{PPV + TPR} = \frac{2TP}{2TP + FP + FN}$	Fowlkes–Mallows index (FM) $= \sqrt{PPV \times TPR}$	Matthews correlation coefficient (MCC) $= \sqrt{TPR \times TNR \times PPV \times NPV} - \sqrt{FNR \times FPR \times FOR \times FDR}$	Threat score (TS), critical success index (CSI), Jaccard index $= \frac{TP}{TP + FN + FP}$

https://en.wikipedia.org/wiki/Confusion_matrix

Many different metrics … we'll dive into a few now, but not all

Critical Idea: Accounting for Thresholds

Remember that, e.g., logistic regression predicts a continuous value, and then we threshold

.5

score(x) < threshold: -1
otherwise: +1

The threshold is in some ways a *hyperparameter* … we can get different, e.g., accuracies with different thresholds.

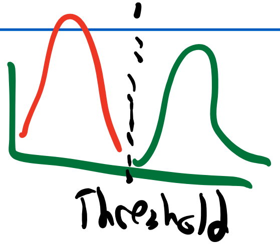
Looking at Thresholds, V1: Precision/Recall Curve

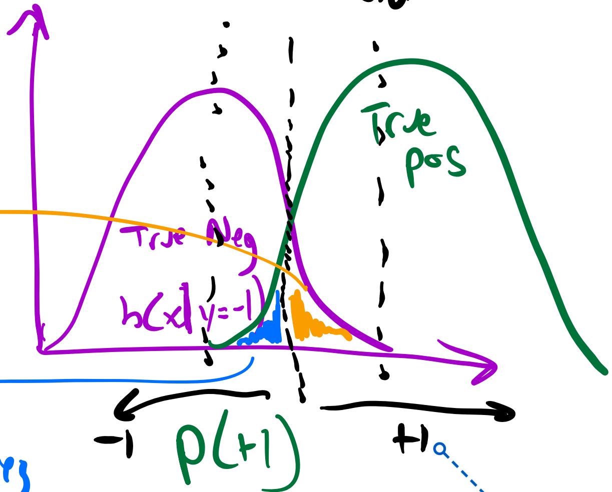
score(x) < threshold : −1

threshold: ∞ ; recall + : 0

recall − : 1

precision + : 0/0 = 1

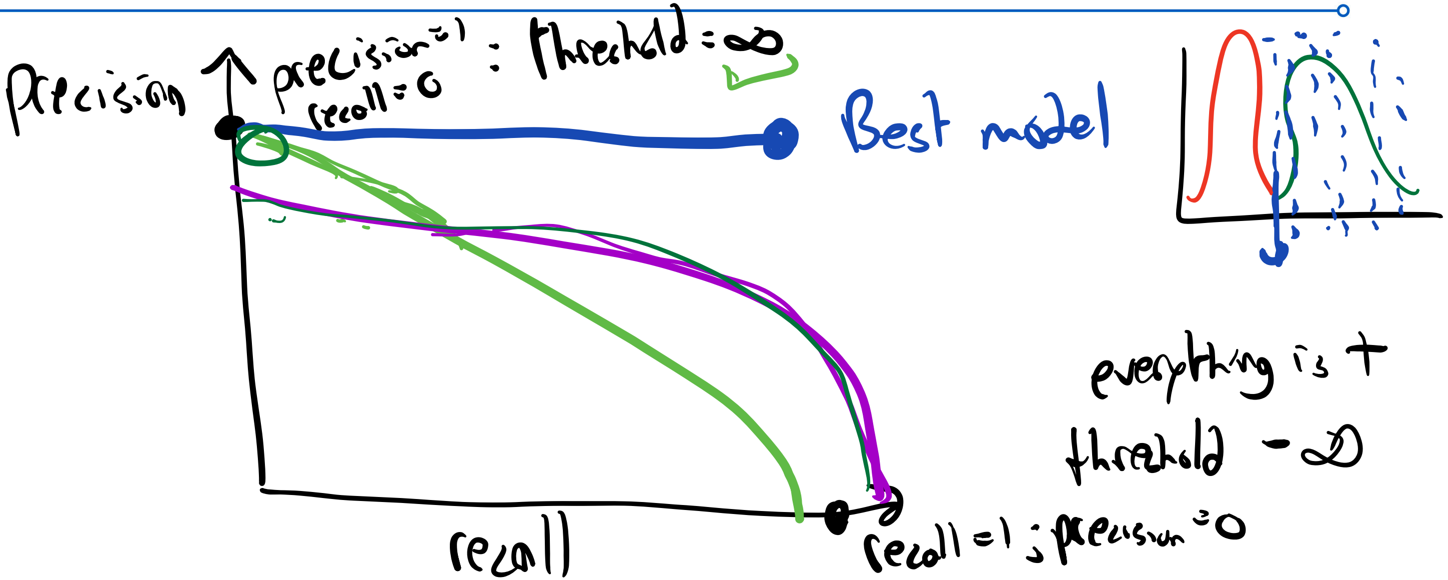




Negatives as positives
("False positives)

True positives as negatives
("False negatives)

Looking at Thresholds, V1: Precision/Recall Curve



- What does the best classifier look like?
- Which is the better classifier?

Looking at Thresholds, V1: Precision/Recall Curve

- How to summarize this?

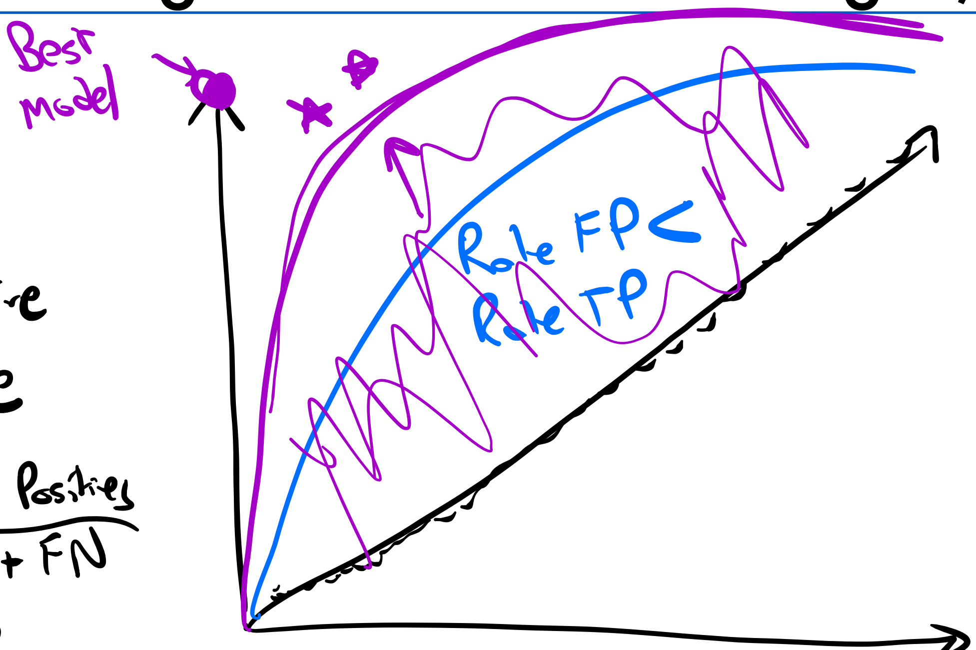
$$\boxed{F_1} = 2 \cdot \frac{\text{precision} \cdot \text{recall}}{\text{precision} + \text{recall}} = \frac{\mathrm{TP}}{\mathrm{TP} + \frac{1}{2}(\mathrm{FP} + \mathrm{FN})}$$

* A metric that unifies precision + recall

Looking at Threshold Changes, V2: ROC

True Positive Rate

$$\frac{\#\text{ True Positives}}{\text{True P} + \text{FN}}$$



False Positive Rate

Looking at Threshold Changes, V2: ROC

- How to summarize this?

$$AUC(f) = \frac{\sum_{t_0 \in \mathcal{D}^0} \sum_{t_1 \in \mathcal{D}^1} \mathbf{1}[f(t_0) < f(t_1)]}{|\mathcal{D}^0| \cdot |\mathcal{D}^1|},$$

Looking at Threshold Changes, V3: Precision @ k

- Final idea: State a number k of observations that you care about, look at precision there
- **Where might this be useful?** search!

① Rank predictions $p(y|x)$

↑ predicted probability

② Pick k := 10

③ $\underbrace{\text{What \% of the top k are +}}_{\text{precision @k}}$

Which metric do we want?

- Diagnosing cancer Recall
- Putting someone in jail Precision

Evaluation Review

- Big ideas:
  - Different metrics for different things
  - Evaluation metrics != loss function
  - Beware of class imbalances
  - Use a lot of metrics!
  - But ultimately, the right metric is tied to your application area

What is missing from these evaluations?

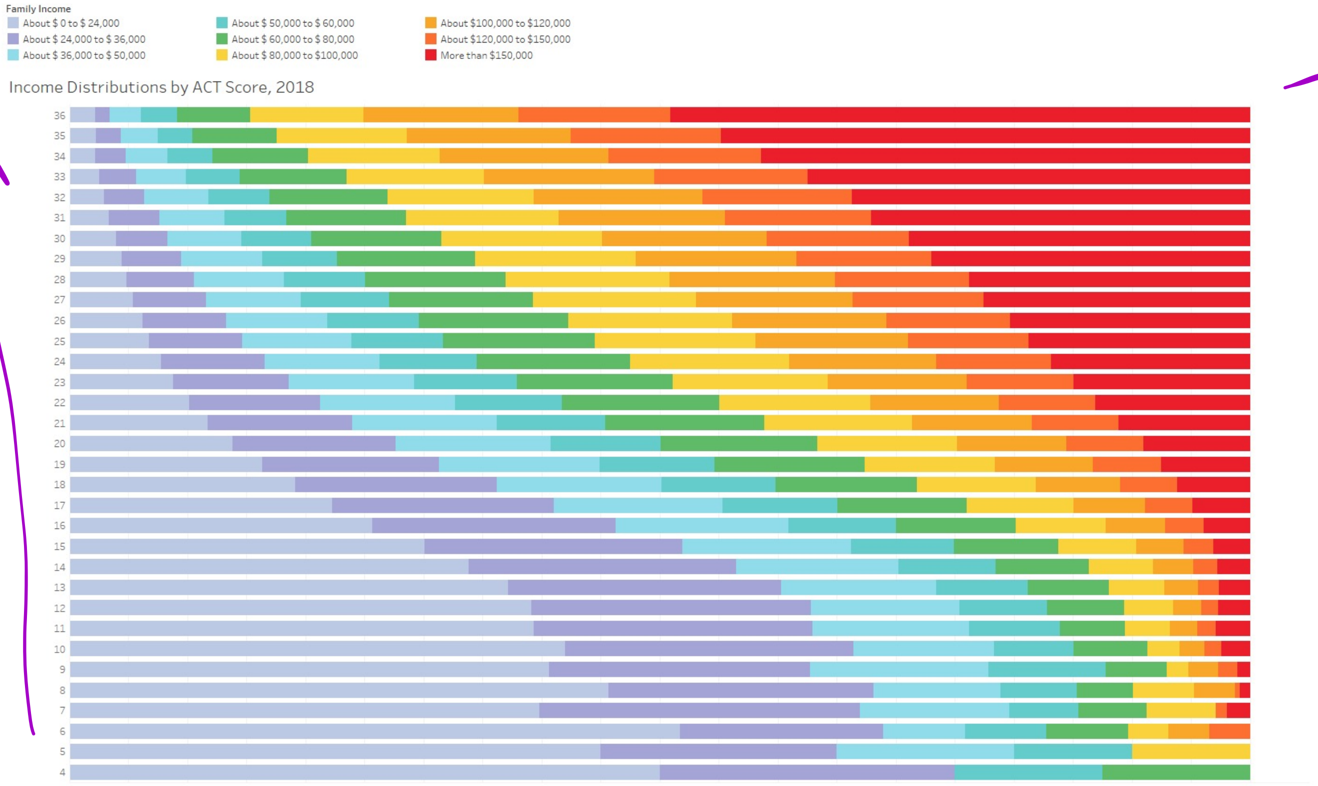
Adding a new feature: height

Geoffrey Hinton
@geoffreyhinton

Suppose you have cancer and you have to choose between a black box AI surgeon that cannot explain how it works but has a 90% cure rate and a human surgeon with an 80% cure rate. Do you want the AI surgeon to be illegal?

3:37 PM · Feb 20, 2020 · Twitter Web App

1,125 Retweets **615** Quote Tweets **5,065** Likes



https://twitter.com/JonBoeckenstedt/status/1447584690932629511/photo/1

Annotation

Annotation Discussion - Overview

- Where do annotations come from?
- How do you know if they’re any good?
  - Accuracy on downstream “expert annotated” data
  - Agreement
    - Percent agreement
    - Krippendorf
- Can we do annotation differently?
  - Aggregation models
  - Snorkel, etc.
  - Considering annotator demographics

This class

Where does data come from?

- Ultimately, most datasets come from *people*
- What might be problematic about that?

Where do *annotations* come from?

"Expert" Annotators (e.g. domain experts)

slow but accurate

≈

amazon mechanical turk™
Artificial Artificial Intelligence

CrowdFlower

…

Challenges with crowd annotation

- How can you incentivize good-faith labels?
- How do you know that you're getting good faith labels?
- How do you aggregate responses across a bunch of people?

Incentivizing Good-faith Labels

- Treat people with respect
  - Pay them
  - Be nice to them

Ensuring Good-faith Labels

- Gold standards – have some observations you know the answer to
- Attention checks – have some questions like "are you awake"
- Redundancy – make sure multiple annotators per observation
- Really, redundancy + **agreement statistics**

Agreement Statistics

- Pairwise agreement: basically, accuracy per annotator

A	B
0	1
0	1
0	0
0	0

% of observations where they agree

Class imbalanced!

- Krippendorf's Alpha

Krippendorf's Alpha

Simpledorff

	document_id	annotator_id	annotation
0	1	B	1
6	3	B	2
7	3	C	2
9	4	B	1
10	4	C	1

+/0/-

annotator_id	1	2	3	4	5	6	7	8	9	10	11	12
A	1	2	3	3	2	1	4	1	2	nan	nan	nan
B	1	2	3	3	2	2	4	1	2	5	nan	3
C	nan	3	3	3	2	3	4	2	2	5	1	nan
D	1	2	3	3	2	4	4	1	2	5	1	nan

$1 - \frac{D_o}{D_e}$

Calculate (1-) the ratio between:

Do – observed disagreements
De – disagreement by chance

https://www.lighttag.io/blog/krippendorffs-alpha/

Krippendorf's Alpha (cont.)

Tweet

annotator_id	1	2	3	4	5	6	7	8	9	10	11	12
A	1	2	3	3	2	1	4	1	2	nan	nan	nan
B	1	2	3	3	2	2	4	1	2	5	nan	3
C	nan	3	3	3	2	3	4	2	2	5	1	nan
D	1	2	3	3	2	4	4	1	2	5	1	nan

Pro +

Neutral

Anti

	1	2	3	4	5	6	7	8	9	10	11	12
1	3	0	0	0	0	1	0	3	0	0	2	**0**
2	0	3	0	0	4	1	0	1	4	0	0	**0**
3	0	1	4	4	0	1	0	0	0	0	0	**1**
4	0	0	0	0	0	1	4	0	0	0	0	**0**
5	0	0	0	0	0	0	0	0	0	3	0	**0**

#annotators

https://www.lighttag.io/blog/krippendorffs-alpha/

Krippendorf's Alpha - observed

Rc

	1	2	3	4	5	6	7	8	9	10	11	12
1	3	0	0	0	0	1	0	3	0	0	2	**0**
2	0	3	0	0	4	1	0	1	4	0	0	**0**
3	0	1	4	4	0	1	0	0	0	0	0	**1**
4	0	0	0	0	0	1	4	0	0	0	0	**0**
5	0	0	0	0	0	0	0	0	0	3	0	**0**

disagreements = 0

disagreements = 3

https://www.lighttag.io/blog/krippendorffs-alpha/

Krippendorf's Alpha – by chance

	1	2	3	4	5	6	7	8	9	10	11	12
1	3	0	0	0	0	1	0	3	0	0	2	**0**
2	0	3	0	0	4	1	0	1	4	0	0	**0**
3	0	1	4	4	0	1	0	0	0	0	0	**1**
4	0	0	0	0	0	1	4	0	0	0	0	**0**
5	0	0	0	0	0	0	0	0	0	3	0	**0**

v_1
v_2
v_3
v_4
v_5

$v_1 \cdot v_2 + v_1 \cdot v_3 \cdots$

https://www.lighttag.io/blog/krippendorffs-alpha/

Krippendorf's Alpha – simple, worked through

Items judged:	1	2	3	4	5	6	7	8	9	10
Meg:	0	1	0	0	0	0	0	0	1	0
Owen:	1	1	1	0	0	1	0	0	0	0

Values:	**0**	**1**	
0	o_{00}	o_{01}	n_0
1	o_{10}	o_{11}	n_1
Number of Values:	n_0	n_1	$n=2N$

	0	**1**	
0	10	4	14
1	4	2	6
	14	6	20

④ Compute **α-reliability** (most simple form): $_{\text{binary}}\alpha = 1 - \frac{D_o}{D_e} = 1 - (n-1)\frac{o_{01}}{n_0 \cdot n_1}$

In the example: $_{\text{binary}}\alpha = 1 - (20-1)\frac{4}{14 \cdot 6} = 0.095$

https://repository.upenn.edu/cgi/viewcontent.cgi?article=1043&context=asc_papers

Aggregation

- The most common approach is **majority vote**
- More recently, people have come up with better ways

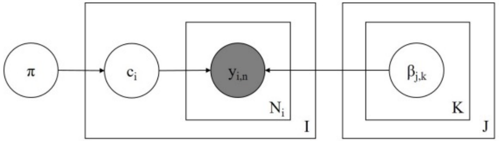


Figure 2: Plate diagram of the Dawid and Skene model.

1979

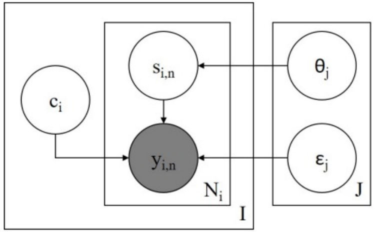


Figure 3: Plate diagram for the MACE model.

https://watermark.silverchair.com/tacl_a_00040.pdf

University at Buffalo
Department of Computer Science and Engineering
School of Engineering and Applied Sciences

Moving forward: Smarter Annotation...

- **Data Programming & Weak Supervision**
- **Data Augmentation**
- **Self-Supervision**
- **Data Selection**

NLP

- More: https://github.com/HazyResearch/data-centric-ai

Active learning