Boosting (almost) by hand

Magic: $$\alpha_t = \frac{1}{2}\ln\left(\frac{1-\epsilon_t}{\epsilon_t}\right)$$

- Initialize weights to uniform dist: $D_1(j) = 1/N$
- For t = 1…T

  Focused on parts that have high weights

  - Train weak learner h_t on distribution D_t over the data
  - Choose weight α_t ← Magic of taking derivative & setting to ∅
  - Update weights:

    for each j

    new weight, old weight

    $$D_{t+1}(j) = \frac{D_t(j)\exp(-\alpha_t y^j h_t(x^j))}{Z_t}$$

    suppose $\alpha_t > 0$

    if h_t correct on j

    $y^j h_t(x^j) > 0$

    ⇒ weight decrease

    if h_t incorrect on j

    weight increases

    - Where Z_t is normalizer:

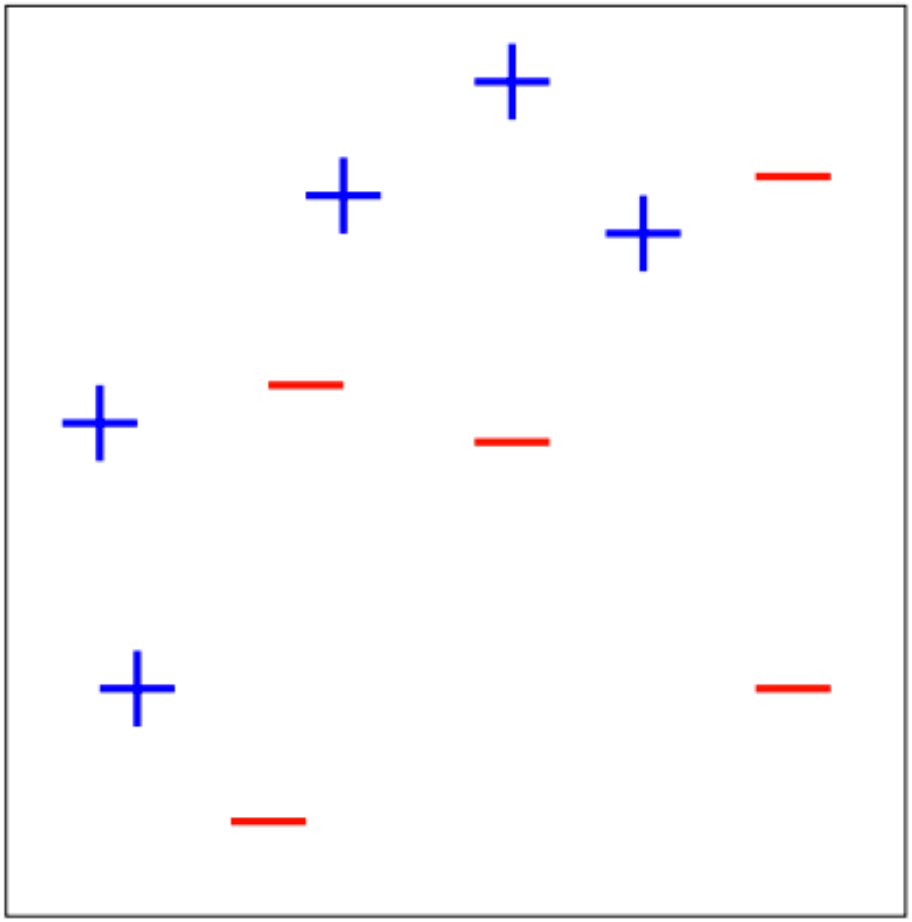
      $$Z_t = \sum_{j=1}^{N} D_t(j)\exp(-\alpha_t y^j h_t(x^j))$$

      so weights add up to 1

- Output final classifier:

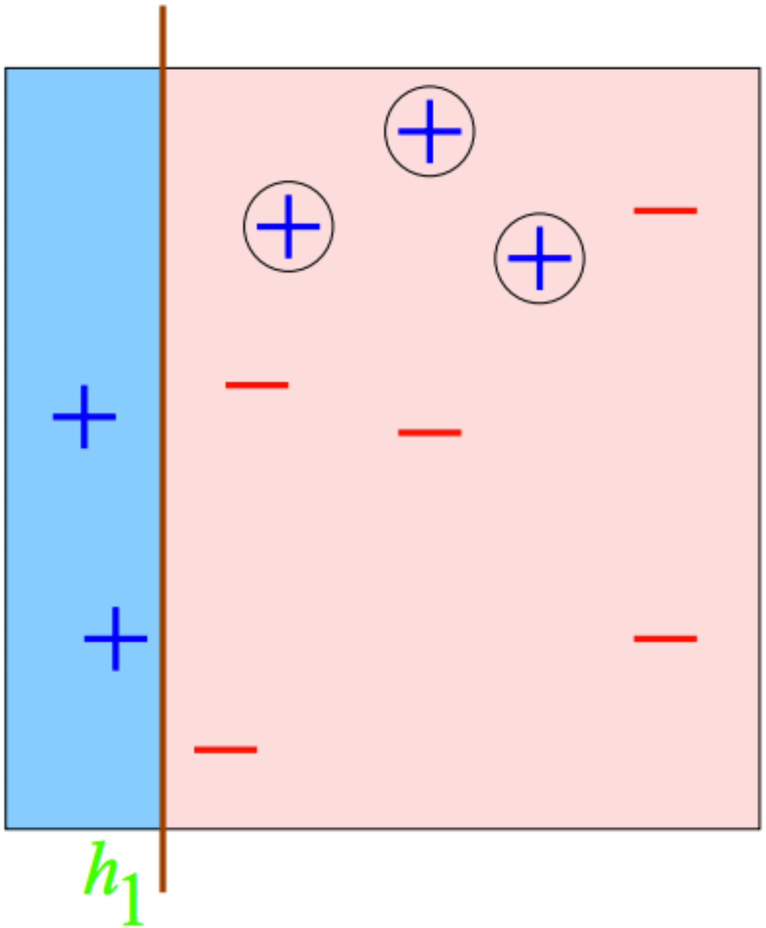
$$H(x) = \text{sign}\left(\sum_{t=1}^{T} \alpha_t h_t(x)\right)$$

(from Rob Schapire)



N = 10

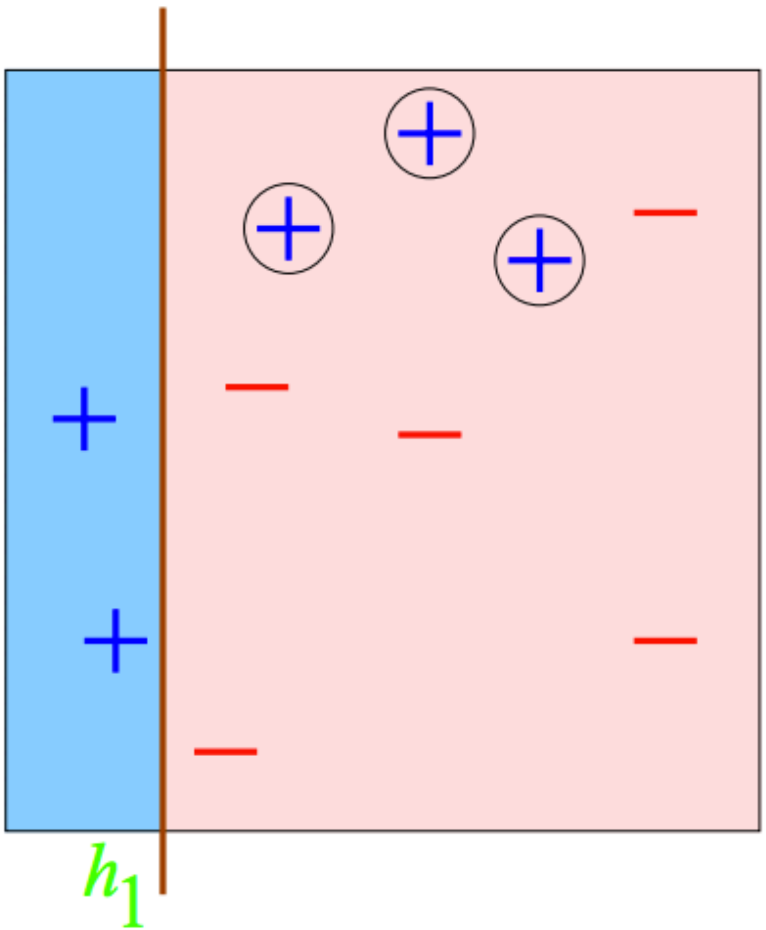
Whats error train here?



$$D_1(j) = \frac{1}{N} = 0.1$$

$$\epsilon_t = \sum_{j=1}^{N} D_t(j) 1[sign(h_t(x^j) \neq y^j]$$

Whats error train here?



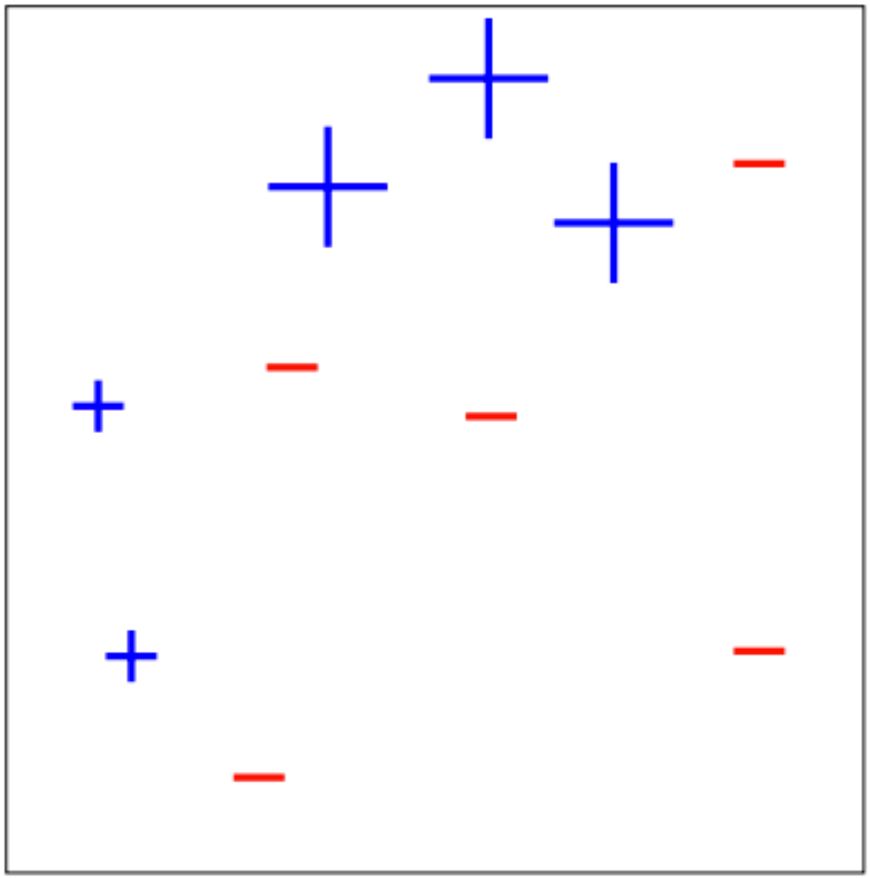
$$D_1(j) = \frac{1}{N} = 0.1$$

$$\epsilon_t = \sum_{j=1}^{N} D_t(j) 1[sign(h_t(x^j) \neq y^j]$$

$$\epsilon_1 = 0.3$$

$$\alpha_1 = \frac{1}{2} ln(\frac{0.7}{0.3}) \approx 0.42$$

New weights

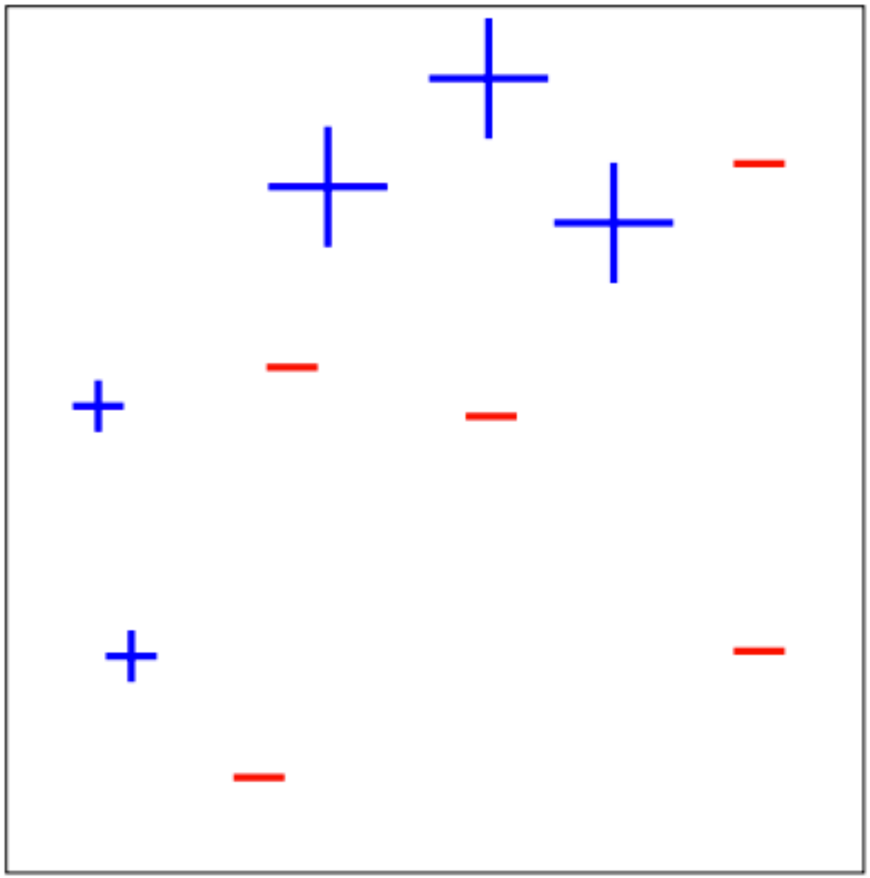


$$D_{t+1}(j) = \frac{D_t(j)\exp(-\alpha_t y^j h_t(x^j))}{Z_t}$$

When is $y^j h_t(x^j) = 1$?

When is $y^j h_t(x^j) = -1$?

New weights

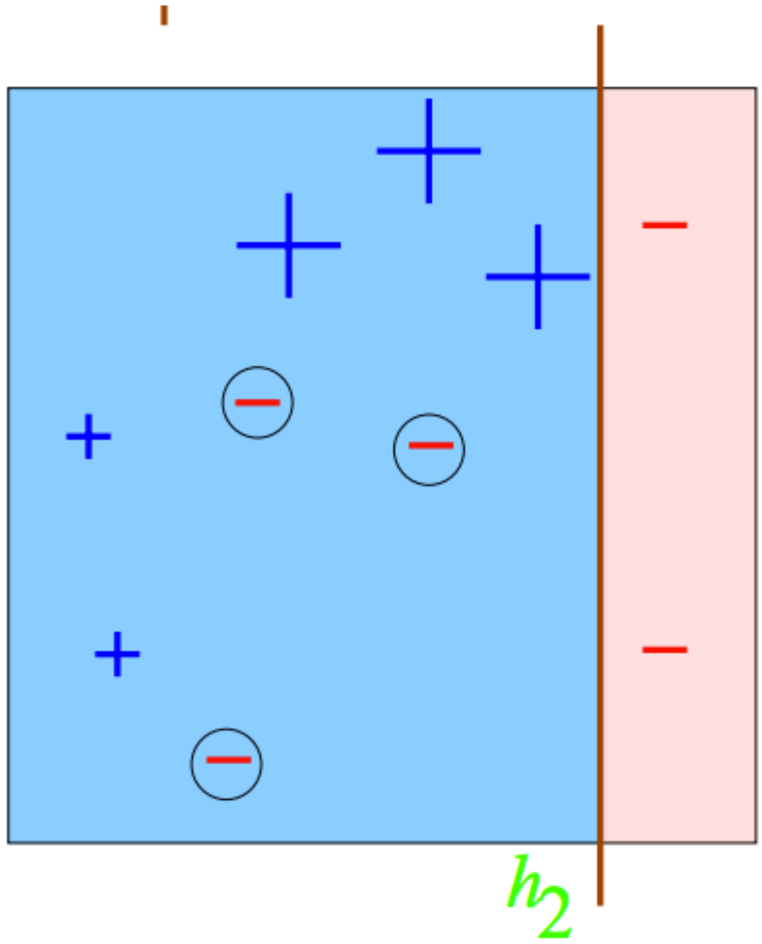


$$D_{t+1}(j) = \frac{D_t(j)\exp(-\alpha_t y^j h_t(x^j))}{Z_t}$$

$$D_2(right) = \frac{0.1exp(-0.42)}{Z_t} \approx 0.071$$

$$D_2(wrong) = \frac{0.1exp(0.42)}{Z_t} \approx 0.166$$

Step 2

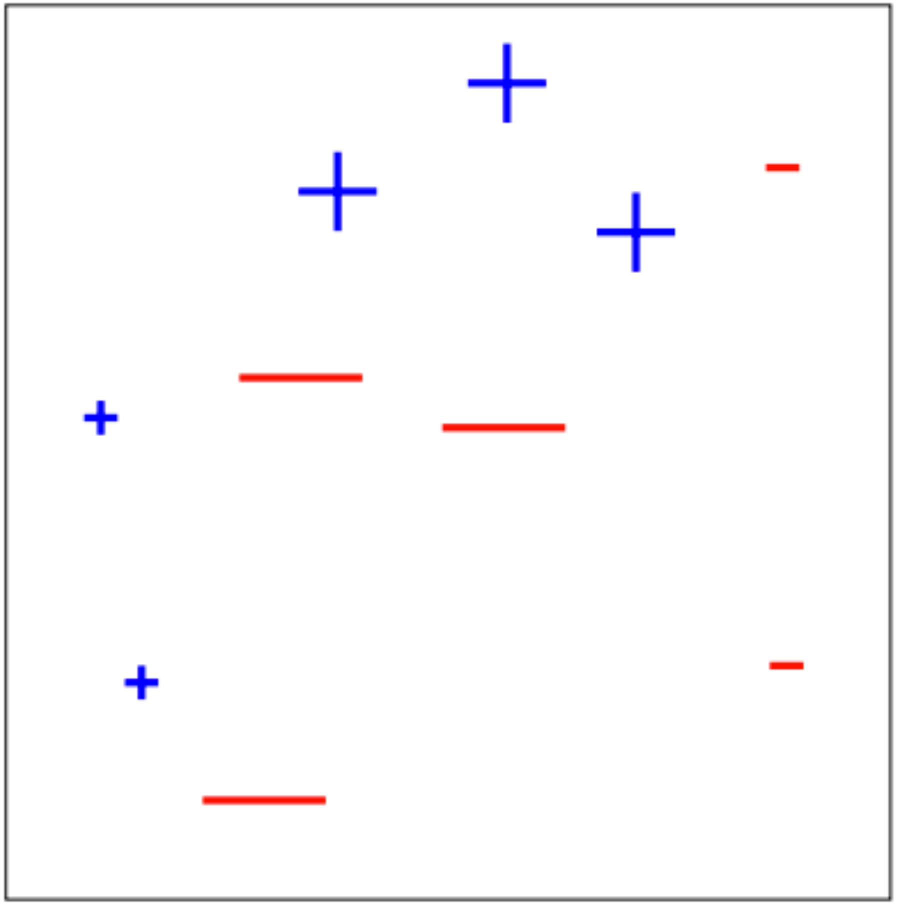


$$\epsilon_2 = 0.071 * 3 \approx 0.21$$

$$\alpha_2 = \frac{1}{2} ln(\frac{0.79}{0.21}) \approx 0.65$$

Notice I still get 3 examples wrong, but they are worth less now.

New weights



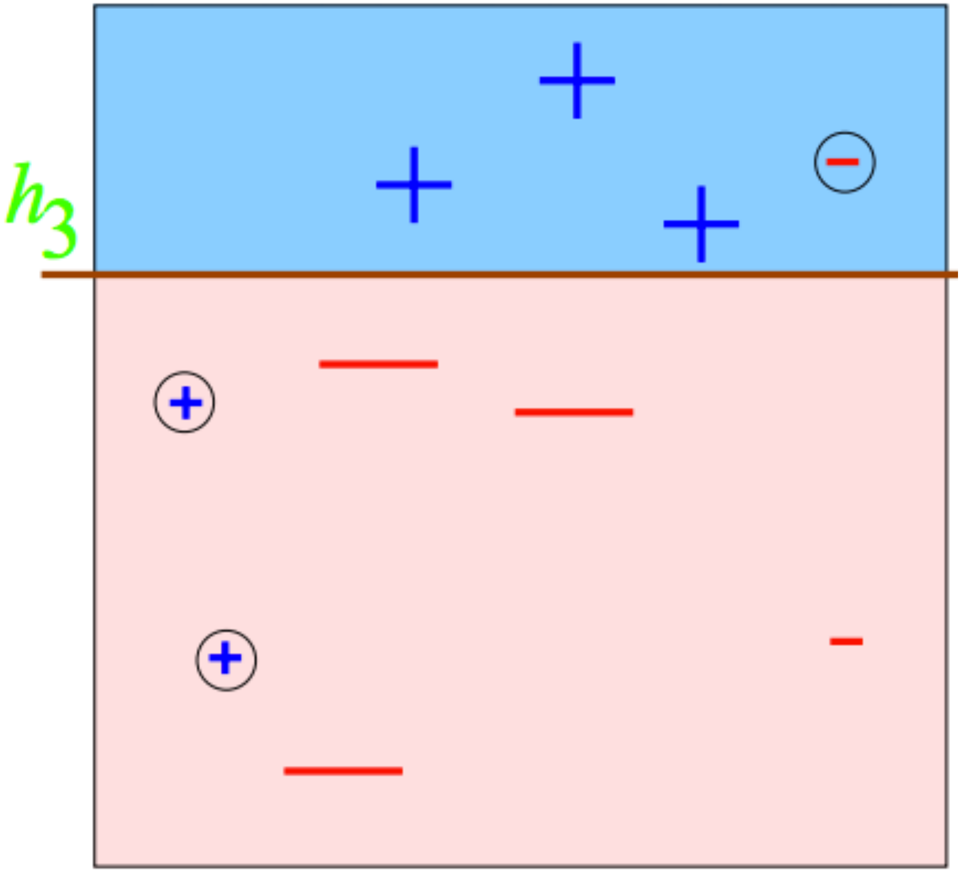
$$D_{t+1}(j) = \frac{D_t(j)\exp(-\alpha_t y^j h_t(x^j))}{Z_t}$$

$$D_3(small + and-) = \frac{0.071exp(-0.65)}{Z_t} \approx 0.045$$

$$D_3(medium+) = \frac{0.166exp(-0.65)}{Z_t} \approx 0.1$$

$$D_3(large-) = \frac{0.166exp(-0.65)}{Z_t} \approx 0.17$$

Step 3



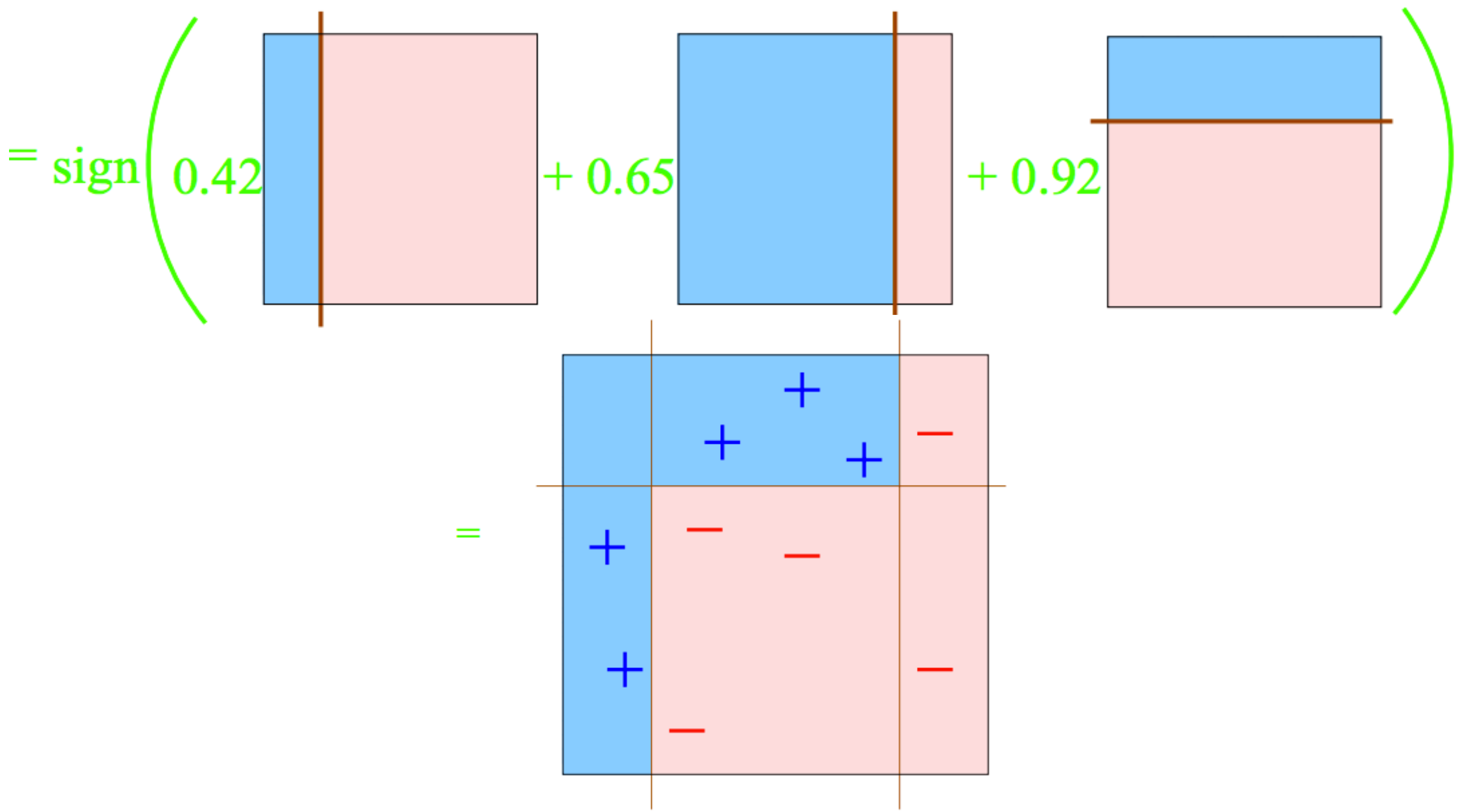
$$\epsilon_3 = 0.045 * 3 \approx 0.14$$

$$\alpha_3 = \frac{1}{2} ln(\frac{0.86}{0.14}) \approx 0.92$$

Notice I still get 3 examples wrong.

- Output final classifier:

$$H(x) = \text{sign}\left(\sum_{t=1}^{T} \alpha_t h_t(x)\right)$$



Evaluation Metrics

- 0-1 error on test set: $$\sum_{j=1}^{N} 1[sign(h_t(x^j) \neq y^j]$$

- 1 – (0-1 error)/N = accuracy

- Accuracy is just % of test samples I get right.

Let's go back to that millionaire…

- Let's say the millionaire asks you to build a classifier to identify other millionaires.

Let’s go back to that millionaire…

- Let’s say the millionaire asks you to build a classifier to identify other millionaires.
- You build a fancy classifier, and get accuracy = 80% in some test data. Pretty good, right?

Let’s go back to that millionaire…

- Let’s say the millionaire asks you to build a classifier to identify other millionaires.
- You build a fancy classifier, and get accuracy = 80% in some test data. Pretty good, right?
- Your competition has a O(1) classifier that has accuracy ~ 99.9%. Should the millionaire fire you right away?

Let's go back to that millionaire…

- Let's say the millionaire asks you to build a classifier to identify other millionaires.
- You build a fancy classifier, and get accuracy = 80% in some test data. Pretty good, right?
- Your competition has a O(1) classifier that has accuracy ~ 99.9%. Should the millionaire fire you right away?
- His classifier just always predicts 'not millionaire'.

Confusion matrix

		Gold standard	
		X	Y
Your Result	X	true positive tp	false positive fp type I error
	Y	false negative fn type II error	true negative tn

Confusion matrix

		Gold standard	
		X	Y
Your Result	X	true positive tp	false positive fp type I error
	Y	false negative fn type II error	true negative tn

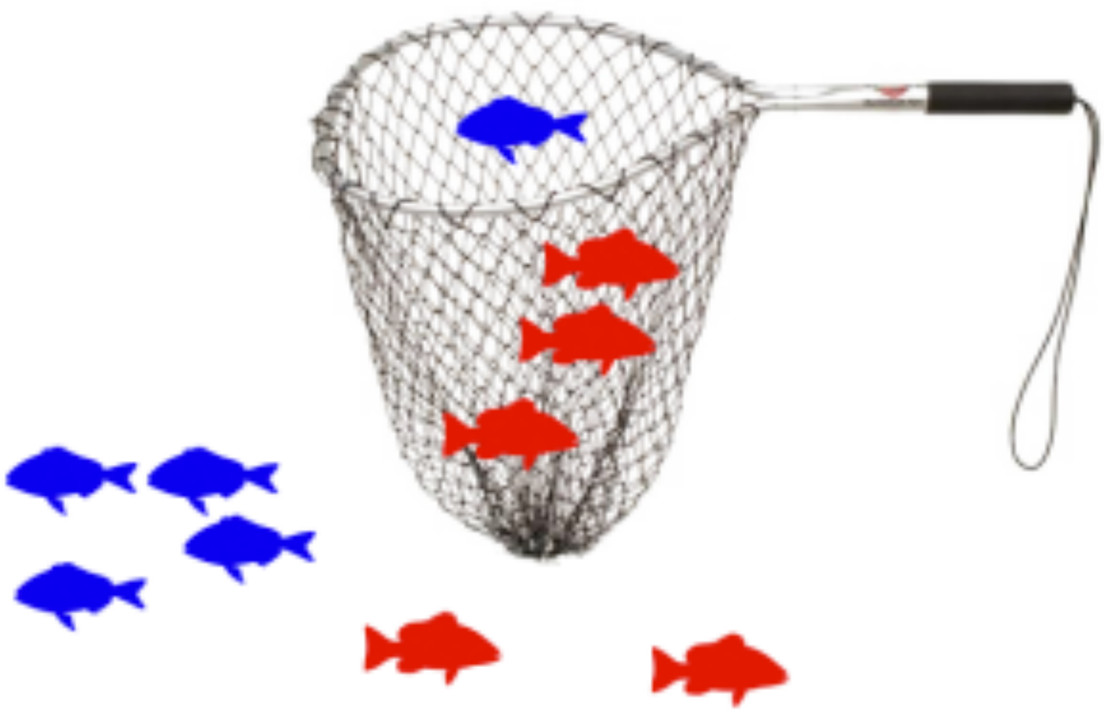
$$accuracy = \frac{tp + tn}{tp + tn + fp + fn}$$

$$precision = \frac{tp}{tp + fp}$$

$$error = \frac{fp + fn}{tp + tn + fp + fn}$$

$$recall = \frac{tp}{tp + fn}$$

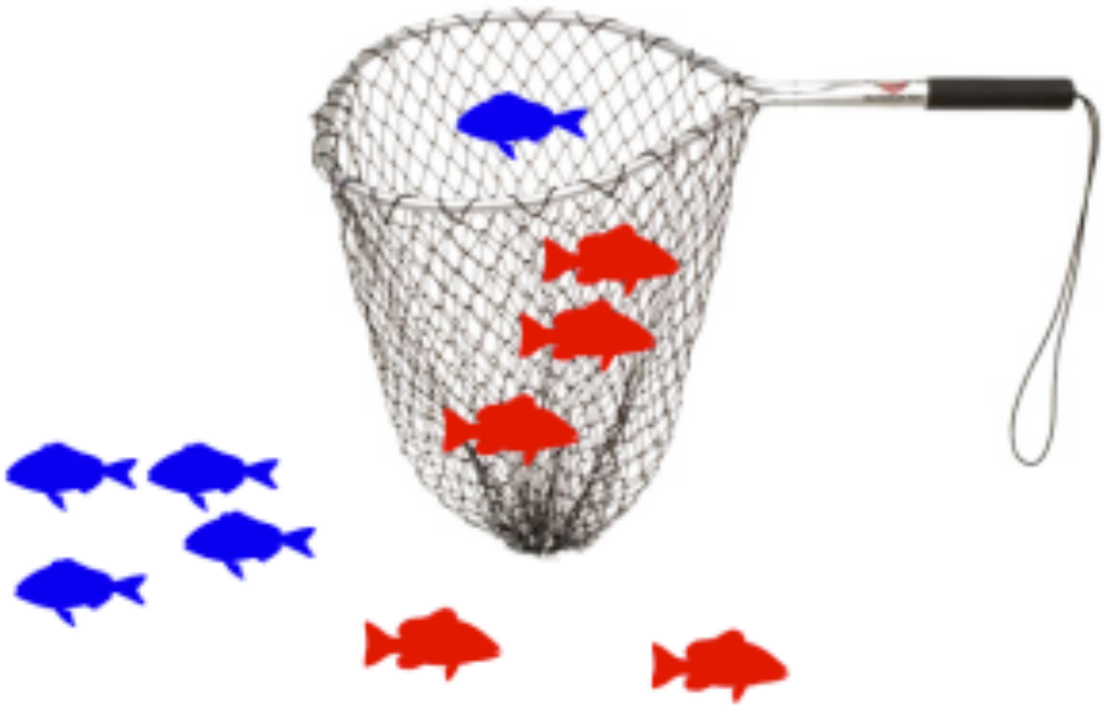
Examples:



Let’s say the net is trying to pick only blue fish. What’s the precision and the recall?

I got this figure from http://www.lucidatainc.com/2012/10/recall-and-precision-understanding-relevancy-in-ediscovery/

Examples:



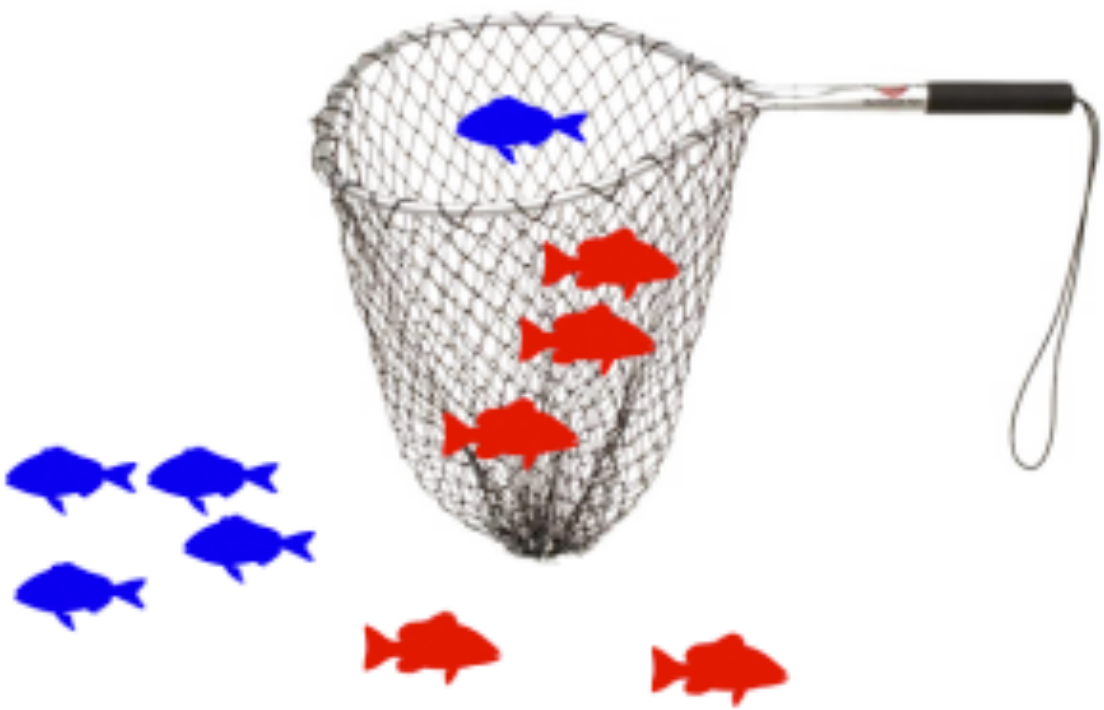
Let's say the net is trying to pick only blue fish. What's the precision and the recall?

P = 1/4

R = 1/5

I got this figure from http://www.lucidatainc.com/2012/10/recall-and-precision-understanding-relevancy-in-ediscovery/

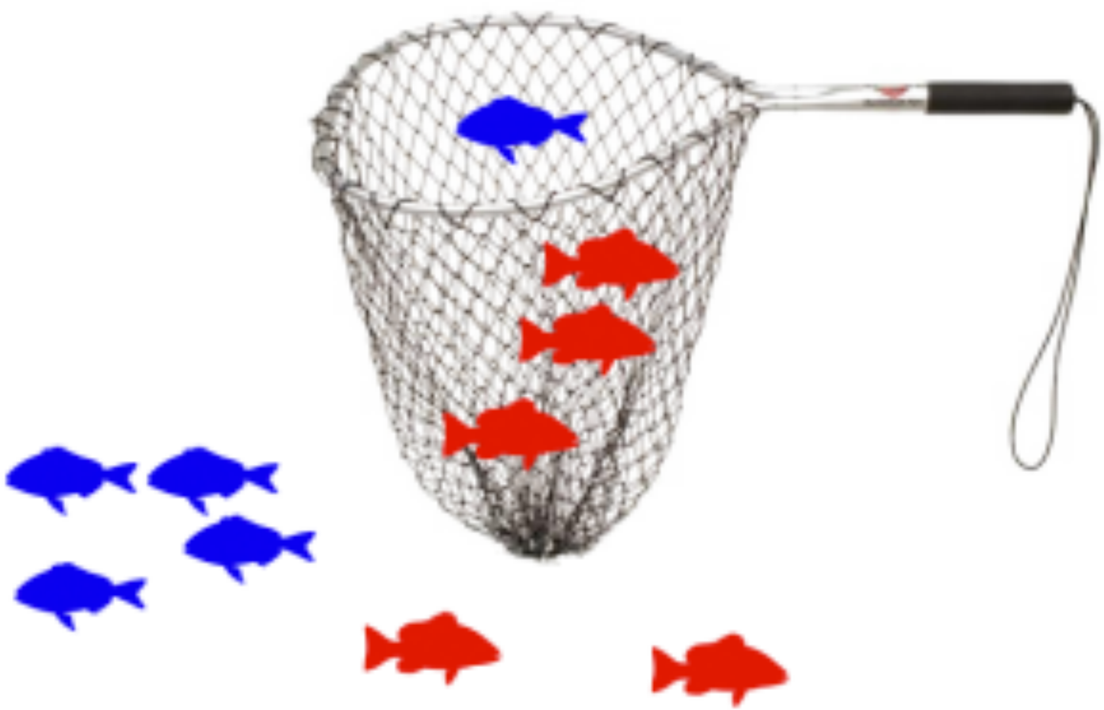
Examples:



Let's say the net is trying to pick only red fish. What's the precision and the recall?

I got this figure from
http://www.lucidatainc.com/2012/10/recall-and-precision-understanding-relevancy-in-ediscovery/

Examples:



Let's say the net is trying to pick only red fish. What's the precision and the recall?

P = 3/4

R = 3/5

I got this figure from
http://www.lucidatainc.com/2012/10/recall-and-precision-understanding-relevancy-in-ediscovery/

A single metric?

- F1 $= \dfrac{2 * \text{precision} * \text{recall}}{\text{precision} + \text{recall}}$
- Fbeta
- AUC
- …

Which one is better?

- Our millionaire identification scenario?

Which one is better?

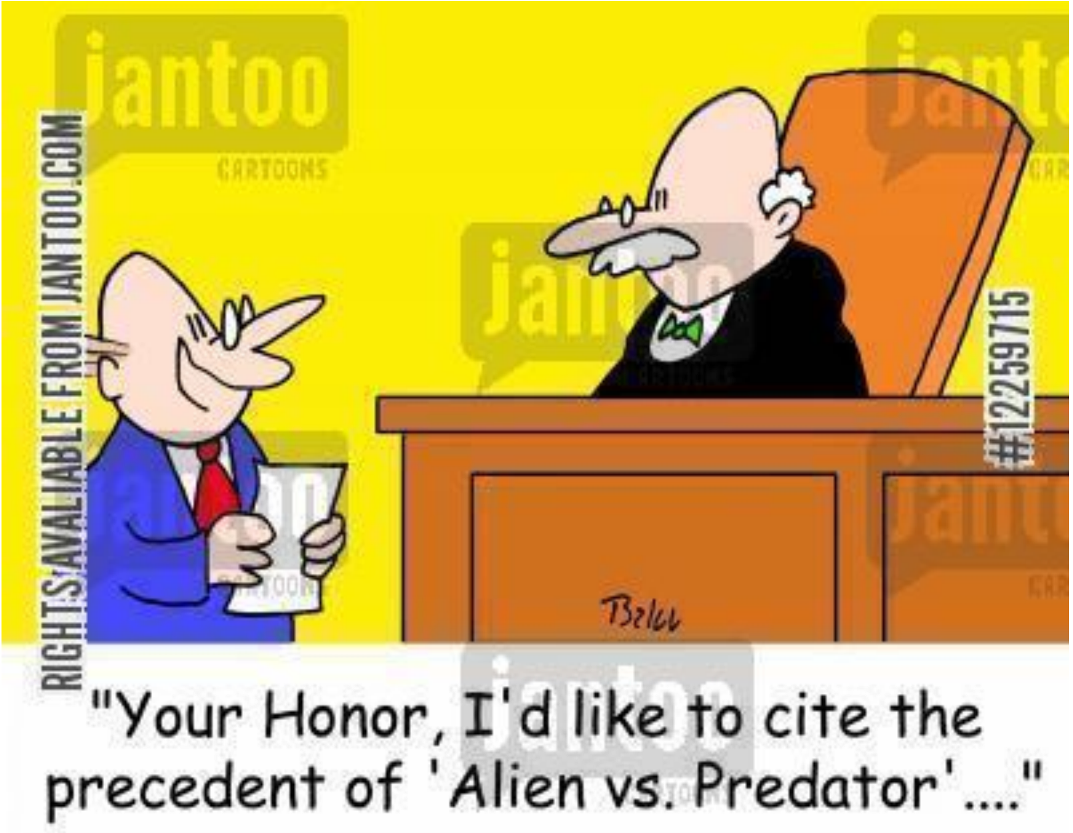
- Spam classification

Which one is better?

- Medical classifier: Y = (operate, don't operate)

Which one is better?

- Search engine: query = legal



Which one is better?

- Search engine: query = "Husky football"
- By the way: why does google show more than 1 page?

Which one is better?

- It depends on the task
- Is there imbalance?
- Are the misclassification costs the same?
- …
- …
- Think about evaluation when doing your projects!