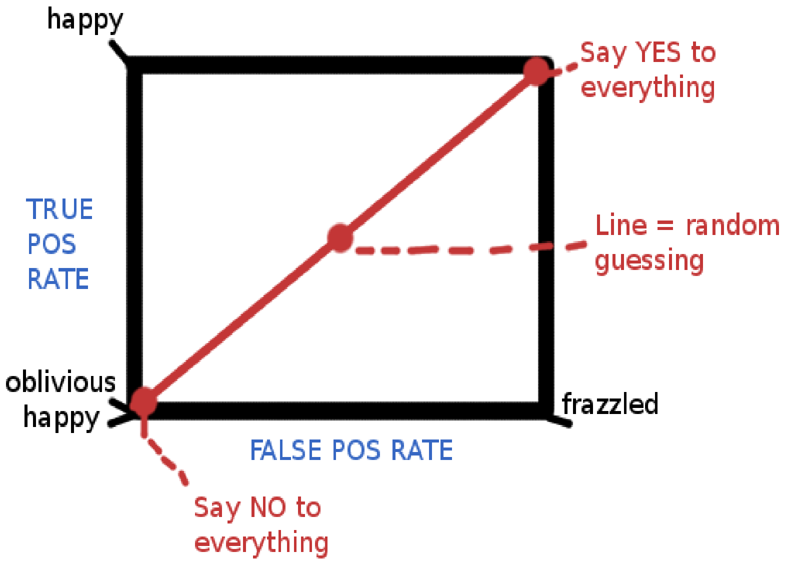
CS424 Machine Learning – 08/12/14

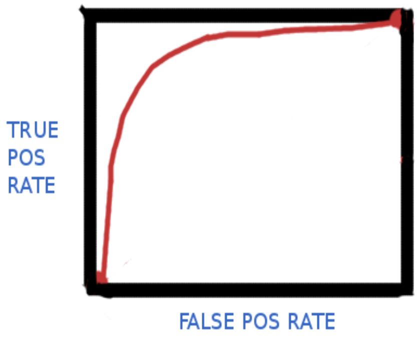
- **ROC Curve (Receiver Operating Characteristics) wiki :**

  Let's use an example...
  Imagine you have a radar that scans the sky to check for aircraft. Each time it senses something it lets out a "Boop!".
  Now imagine we can adjust a threshold that helps decide when a warning bell should sound
  - Low threshold = high sensitivity = the slightest thing will sound the bell
  - High threshold = low sensitivity = we could be bombed in our sleep

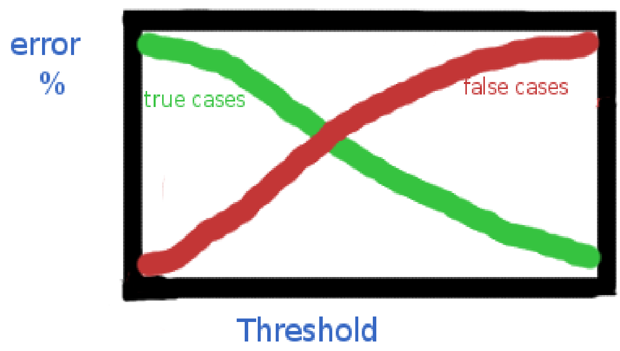
  Evidently there's a trade-off in how good the settings are.



In a reasonably trained system, we should get a curve like this....



(Parametric form)
-------------------->



- **<u>Strong learners vs weak learners</u>**

    - **Strong** *:*

      Given a dataset, we can train it in polynomial time. How big should the training set be?
      Use PAC (Probably Approximately Correct) :
      - Level of error you're willing to tolerate. After training, your error rate should be lower than tolerance E.
        P(P(error off sample) <= E) >= 1 – δ

      Can guarantee this is the case given some number of samples in strong learning. The number of samples needed for P.A criteria = **poly(1/E, 1/ δ)**

      Also look at V.C dimension.

- **Weak:**

  Exists an E such that given a polynomial number of samples, your performance will be better than random guessing for E.
  P(P(error off sample) <= ½ – E) >= δ

- **Can weak learners = strong learners?**
  Weak learner ----> [BOOSTER] -----> Strong learner

  You can train weak learners so that you're sure its doing better than random guesses / chance. Then run new data through this. Take any 'wrong' classified data and put this in a new distribution. Train the weak learners on this (it will boost E). Keep improving E until it approaches ½ so that it matches the 'strong' bound.
  (Also see ADA Boost)