Metrics Part 2

22 January 2026

Alex Lyman

Statistical Significance

Statistical Significance – The Problem

- How do we know that some measurement is statistically significant vs being just a random perturbation
  - How good a predictor of generalization accuracy is the sample accuracy on a test set?
  - Is a particular hypothesis really better than another one because its accuracy is higher on a validation set?
  - When can we say that one learning algorithm is better than another for a particular task or set of tasks?
- Key point – **What is the probability that the differences in our results are just due to chance**?

Confidence Intervals - Intuition

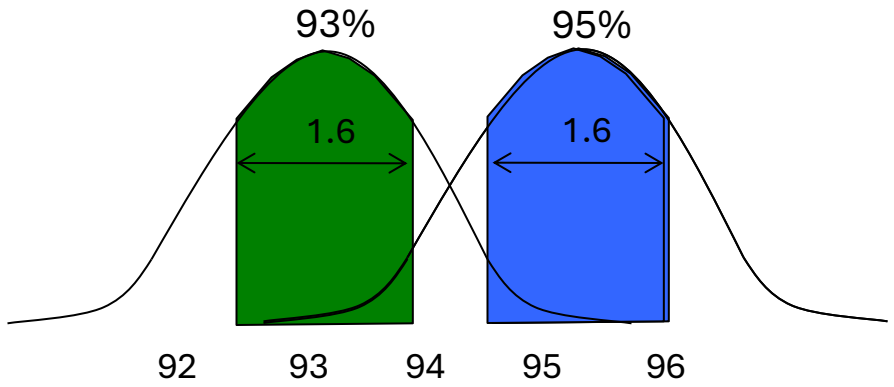
- Which of these archers is better?
- Archer A - average arrow hits 8 ring – little spread
- Archer B – average arrow hits 9 ring – big spread
- If we had each archer take one shot, can we be sure who would be better?
- Even though Archer B has a better center point (average), there is a lot of variance. We can't say for sure whether Archer B is a better archer.

Target A

Target B

Confidence Intervals - Formalized

- An *N*% confidence interval for a parameter *p* is an interval that is expected with probability *N*% to contain *p*.
  - 95% confidence interval – there is a 95% chance that this interval contains the value.
- The true mean (or whatever parameter we are estimating) will fall in the interval $\pm C_N\sigma$ of the sample mean with *N*% confidence.
- σ is the deviation
- C_N gives the width of the interval about the mean that includes *N*% of the total probability under the particular probability distribution
- C_N is a distribution specific constant for different interval widths.
- Statistical significance tests, confidence intervals needed for publications.



Loss

Loss

$$L_{abs}(y_i, \hat{p}_i) = |y_i - \hat{p}_i|$$

A numerical measure of the **error** between a model's **predicted** output and the actual **ground truth** value.

- y_i
  - Ground truth label
- $\hat{p}_i$ Prediction

Loss

- Machine learning algorithms make predictions.
  - We want those predictions to be really good
  - Learning uses differences between the predicted value and the actual value to make adjustments so the model predicts better
- Absolute Loss
  - Simple absolute difference

$$L_{abs}(y_i, \hat{p}_i) = |y_i - \hat{p}_i|$$

  - Works well for single valued data
  - What about multi-class data?
    - Output is probability of being in a class
    - Zybooks uses nondiabetic, prediabetic, diabetic probabilities as example

Cross-Entropy Loss

- Used when the output has more than one feature.
- Compares the difference between the predicted probability distribution and the true distribution
  - View multi-featured output class as a probability distribution

$$L = -\sum_{k=1}^{K} y_k \log(p_k)$$

- Example – predict whether dog, cat, or bird
  - True label – [1, 0, 0] (probability of each class)
  - Predicted – [0.7, 0.2, 0.1]
  - Loss = - (1* log(0.7) + 0*log(0.2) + 0 * log(0.1)) = 0.357
  - Predicted – [0.9, 0.05, 0.05] → loss = 0.105

Precision, Recall, F1

- Precision
  - True positive results divided by the number of all positive results
  - How precise/accurate is your model for predicting positive
  - Use when there is a high cost for False Positive
- Recall
  - The number of true positive results divided by the number of all samples that should have been identified as positive
  - Percentage of the actual positives we predict correctly
  - Use when there is a high cost for False Negative
- F1
  - The harmonic mean of Precision and Recall
  - Looking at the balance between Precision and Recall

Why the harmonic mean?

- Precision and recall both have **true positives** in the **numerator**, but they have **different denominators**. To average them it only makes sense to average their reciprocals, thus the harmonic mean.

ACTUAL VALUES	POSITIVE	NEGATIVE
POSITIVE	TP	FN
NEGATIVE	FP	TN

$$Precision = \frac{TP}{TP + FP} \qquad Recall = \frac{TP}{TP + FN}$$

$$Accuracy = \frac{TP + TN}{TP + FP + FN + TN}$$

$$F1\ Score = 2 \times \frac{Precision \times Recall}{Precision + Recall}$$

Harmonic Mean – What is it?

- The harmonic mean can be expressed as the *reciprocal of the arithmetic mean of the reciprocals of the given set of observations*.
- Example, the harmonic mean of 1, 4, and 4 is:

$$\left(\frac{1^{-1}+4^{-1}+4^{-1}}{3}\right)^{-1} = \frac{3}{\frac{1}{1}+\frac{1}{4}+\frac{1}{4}} = \frac{3}{1.5} = 2.$$

- For the special case of just two numbers:

$$H = \frac{2x_1 x_2}{x_1 + x_2}$$

Why the harmonic mean?

- What is the average of 30mph and 40mph?
  - If you drive for 1 hour at each speed, the average speed over the 2 hours is the arithmetic average, 35mph.
- However, if you drive for the same distance at each speed -- say 10 miles -- then the average speed over 20 miles is the *harmonic mean* of 30 and 40, about 34.3mph.
- The reason is that for the average to be valid, you really need the values to be in *the same scaled units*. Miles per hour needs to be compared over the same number of hours; to compare over the same number of miles you need to average hours per mile instead, which is exactly what the harmonic mean does.

$$H = \frac{2x_1 x_2}{x_1 + x_2} \qquad \frac{2 * 30 * 40}{30 + 40} = \frac{2400}{70} = 34.285$$

Harmonic Mean Continued

- *The reciprocal of the arithmetic mean of the reciprocals of the given set of observations*

$$\frac{2}{\frac{1}{30}+\frac{1}{40}} \quad = 2 \div \frac{7}{120} = 34.285$$

Harmonic Mean

$$\text{F1 Score} = \frac{2}{\frac{1}{\text{Precision}} + \frac{1}{\text{Recall}}} = \frac{2 \times \text{Precision} \times \text{Recall}}{\text{Precision} + \text{Recall}}$$

$$H = \frac{2x_1 x_2}{x_1 + x_2}$$

F_β Score

F_β Score (F-Beta Score)

- F1 score weights Precision and Recall the same
- What if we wanted to take both into account, but *give one more weight?*

$$F_\beta = (1 + \beta^2) \cdot \frac{\text{Precision} \cdot \text{Recall}}{(\beta^2 \cdot \text{Precision}) + \text{Recall}}$$

- This equation is the F1 score when Beta = 1

F_β Score (F-Beta Score)

- **β=1:** Balanced (F1). Precision and Recall are equal.

- **β>1 (e.g., 2):** Favor Recall
  - *Example:* Cancer Screening.
  - Missing a sick patient is fatal. False alarms are not.
  - *Use:* F2 Score.

- **β<1 (e.g., 0.5):** Favor Precision
  - *Example :* YouTube Recommendations. Showing a bad video makes users leave. Missing a good video is probably ok, since there are lots.
  - *Use:* F0.5 Score.

$$F_\beta = (1 + \beta^2) \cdot \frac{\text{Precision} \cdot \text{Recall}}{(\beta^2 \cdot \text{Precision}) + \text{Recall}}$$

Multi-Class F Score

Multi-Class F Score

- Precision/Recall/F1 are binary metrics. They only know "Yes" and "No."
- How do we apply them to a dataset with 3 Classes (A, B, C)?
- **"One-vs-All" Approach** We break the problem into 3 separate binary problems:
  - **Is it A?** (A vs. Not-A) → Calculate F1(A)
  - **Is it B?** (B vs. Not-B) → Calculate F1(B)
  - **Is it C?** (C vs. Not-C) → Calculate F1(C)
- Then what?

Multi-Class F Score – Macro Averaging

- Calculate the metric for each class independently, then take the Simple Average.

$$\text{Macro F1} = \frac{F_1(A) + F_1(B) + F_1(C)}{3}$$

- Every class gets an equal vote, regardless of how much data it has.
- Use this when you care about performance on **Rare Classes**.
  - *Example:* Detecting rare diseases. If you have 1,000 healthy patients and 10 sick ones, and you miss all the sick ones:
  - Macro-F1 will be **Low** (because the "Sick" class score is 0). This correctly alerts you to the failure.

Multi-Class F Score – Micro Averaging

- Pool all the True Positives, False Positives, and False Negatives from all classes into one giant bucket first, then calculate the metric once.

$$\text{Micro Precision} = \frac{\text{Total TP}}{\text{Total TP} + \text{Total FP}}$$

- Use this when you care about **Overall Accuracy** or total throughput. (Rare disease has costly treatment that's not necessary)
  - *Example:* If you miss all the rare "Sick" patients but get all the 1,000 "Healthy" ones right:
  - Micro-F1 will be **High** (approx 99%). It hides the failure on the rare class.

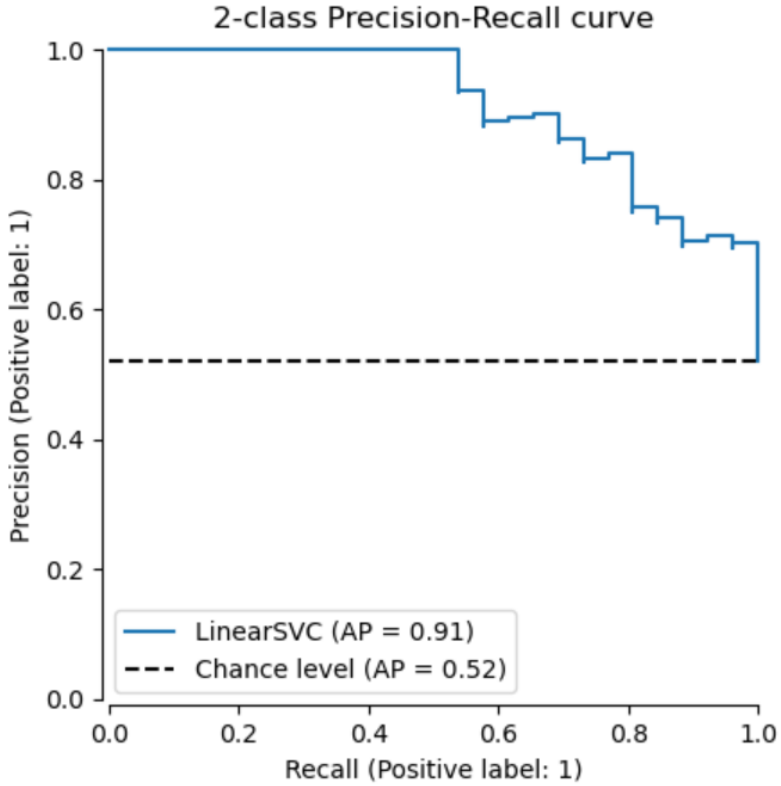
Multi-Class F Score – Review

- F-score is binary, so if we have multiple classes, we need to do something special.
- One-vs-All approach
- How to combine?
- Macro Averaging (US Senate) weights all classes equally. (Good for rare classes)
- Micro Averaging (US House of Reps) weights according to prevalence in data (good for overall performance)

Precision-Recall Curve

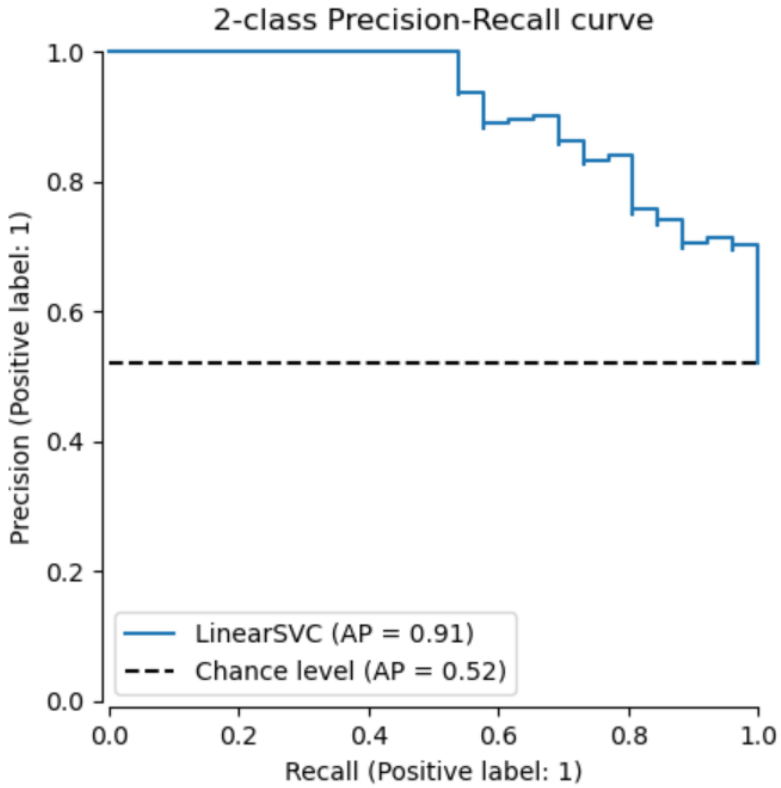
Precision-Recall Curve

- Classifiers don’t output a yes/no, they output a class probability.
- We use a decision threshold to determine how to classify.
- Moving the decision boundary creates tradeoff between Precision and Recall.
- We can plot this (kind of like an ROC curve, but you have to know the difference between them)



Precision-Recall Curve

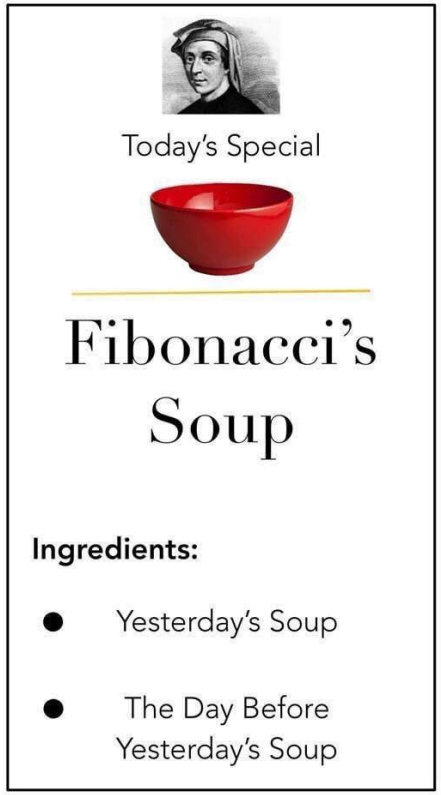
- Recall on X axis, Precision on Y
- Top RIGHT is informally the best point
- Random chance is a *horizontal* line
- Still use AUC to compare
- Choose different points on the curve to balance precision and recall
- Why doesn't this example curve look smooth?



Precision-Recall Curve VS ROC Curve

- Both visualize performance across **all possible thresholds** (from 0.0 to 1.0)
- **Recall (True Positive Rate)** is in both!
  - **ROC:** Y-axis
  - **PR:** X-axis
- Denominator is different
  - **ROC:** Divide by Real Negatives (Usually Large)
  - **PR:** Divide by Predicted Positives (Usually Small)
- When might you use?
  - **ROC:** Classes are Balanced
  - **PR:** You have a "Needle in a Haystack" (e.g., Fraud: 1% vs. 99%).

Quiz Time


Today's Special
Fibonacci's Soup
Ingredients:
Yesterday's Soup
The Day Before Yesterday's Soup