Evaluation Metrics

Astrini Sie

asie@seattleu.edu

Week 5

ECEGR4750 - Introduction to Machine Learning
Seattle University

October 17, 2023

Recap and Updates

- Lab Take Home Assignment due this Thursday at 11.59pm
- Office Hours
  - T, Th 12-1p at Bannan 224
  - W 7-9p via Zoom
  - F 9-9.45a via Zoom
- Zoom Link: `https://seattleu.zoom.us/j/7519782079?pwd=cnhCM2tPcHJKVWwxZVArS2VHSUNJZz09`
  - Meeting ID: 751 978 2079
  - Passcode: 22498122

- Multi-class Classification
- Exponential Family
- Generalized Linear Models

Overview

1. Binary Classification
    - Accuracy
    - Confusion Matrix
    - Area Under ROC
    - Area Under Precision Recall Curve
    - Log Loss
    - Brier Score
    - Class Imbalance Revisit
2. Multi-Class Classification
3. Regression
4. Task Specific Models
5. Closing Note

Why?

- To know how good the model we designed is performing

Binary Classification

Accuracy

$$Accuracy = \frac{n_{correct\_predictions}}{n_{all\_predictions}}$$

- It seems simple, but it is the most mis-used evaluation metrics!
- Suitable only for the case of balanced classes.

COVID (2%)

Healthy (98%)

For any given input x, the model h always returns $\hat{y} = healthy$. In the sample above, the accuracy:

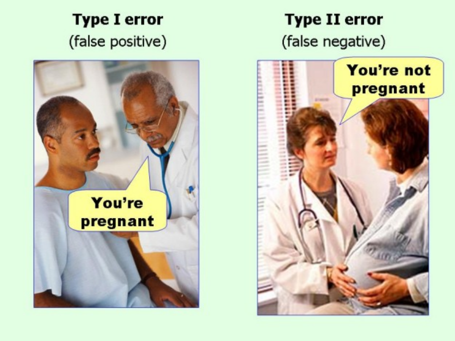
$$Accuracy = \frac{98}{100} = 98\% \quad \rightarrow$$ **Misleading!!!**

Confusion Matrix

		Prediction	
		Positive (PP)	Negative (PN)
Actual	Positive (P)	True Positive (TP)	False Negative (FN)
	Negative (N)	False Positive (FP)	True Negative (TN)

Confusion Matrix

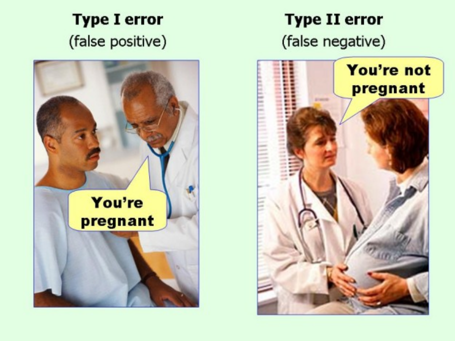
False Positive and False Negative



- In some cases False Positive is "less bad" than False Negative, and vice versa. Can you name some examples?

Confusion Matrix

False Positive and False Negative



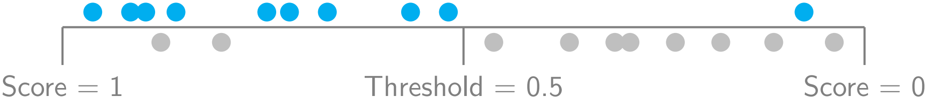
- COVID Test: False Negative is bad! (contagious person went out and spread disease). False Positive is not too bad! (healthy person stuck quarantining for a couple of days)
- Ads Placement: False Positive is bad! (money spent wrongly)

Confusion Matrix

Metrics

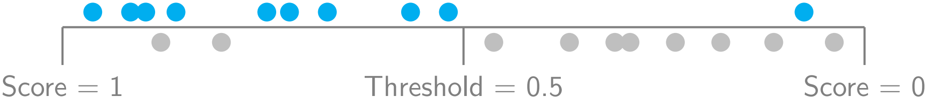
- $Accuracy = \frac{TP+TN}{P+N}$
- $Precision = \frac{TP}{PP}$
  How *precise* the model is right when it says 'yes'
- $Recall$ (Positive Recall / Sensitivity) $= \frac{TP}{P}$
  How often the model says 'yes' when the answer is 'yes'
- $Specificity$ (Negative Recall) $= \frac{TN}{N}$
- $F1 = 2 \cdot \frac{Precision \cdot Recall}{Precision + Recall}$

Confusion Matrix - Sample Case



		Prediction	
		$+$ PP	$-$ PN
Actual	$+$ P	🟩	🟥
	$-$ N	🟥	🟩

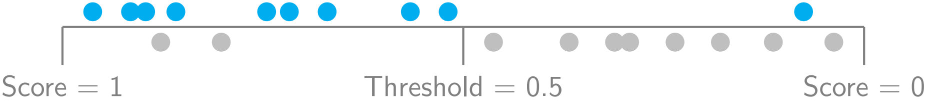
Confusion Matrix - Sample Case



		Prediction	
		$+\ PP$	$-\ PN$
Actual	$+\ P$	$TP = 9$	$FN = 1$
	$-\ N$	$FP = 2$	$TN = 8$

Confusion Matrix - Sample Case

Accuracy



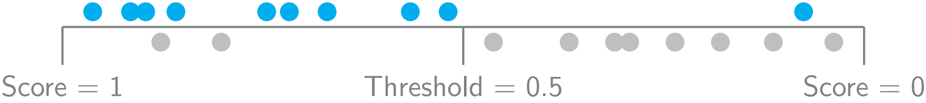
		Prediction	
		$+\ PP$	$-\ PN$
Actual	$+\ P$	$TP = 9$	$FN = 1$
	$-\ N$	$FP = 2$	$TN = 8$

$$Accuracy = \frac{TP + TN}{P + N}$$

$$Accuracy = \frac{9 + 8}{10 + 10} = 0.85$$

Confusion Matrix - Sample Case

Precision



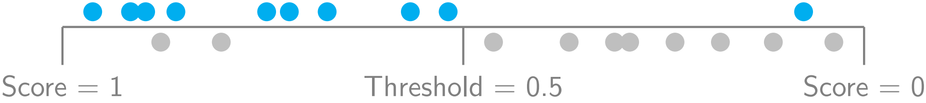
		Prediction	
		$+\ PP$	$-\ PN$
Actual	$+\ P$	$TP = 9$	$FN = 1$
	$-\ N$	$FP = 2$	$TN = 8$

$$Precision = \frac{TP}{PP}$$

$$Precision = \frac{9}{11} = 0.81$$

Confusion Matrix - Sample Case

Recall (Positive Recall / Sensitivity)



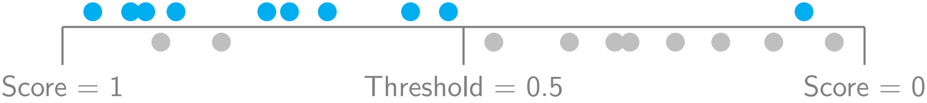
		Prediction	
		$+$ *PP*	$-$ *PN*
Actual	$+$ *P*	$TP = 9$	$FN = 1$
	$-$ *N*	$FP = 2$	$TN = 8$

$$Recall = \frac{TP}{P}$$

$$Recall = \frac{9}{10} = 0.9$$

Confusion Matrix - Sample Case

Negative Recall / Specificity



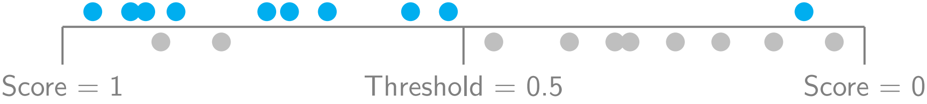
		Prediction	
		$+\ PP$	$-\ PN$
Actual	$+\ P$	$TP = 9$	$FN = 1$
	$-\ N$	$FP = 2$	$TN = 8$

$$Specificity = \frac{TN}{N}$$

$$Specificity = \frac{8}{10} = 0.8$$

Confusion Matrix - Sample Case

F1 Score



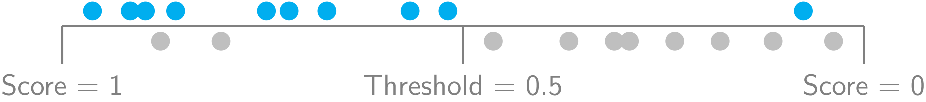
		Prediction	
		+ *PP*	− *PN*
Actual	+ *P*	$TP = 9$	$FN = 1$
	− *N*	$FP = 2$	$TN = 8$

$$F1 = 2 \cdot \frac{Precision \cdot Recall}{Precision + Recall}$$

$$F1 = \frac{0.81 \cdot 0.9}{0.81 + 0.9} = 0.857$$

Confusion Matrix - Sample Case

Summary



		Prediction	
		$+$ *PP*	$-$ *PN*
Actual	$+$ *P*	$TP = 9$	$FN = 1$
	$-$ *N*	$FP = 2$	$TN = 8$

$$Accuracy = 0.85$$

$$Precision = 0.81$$

$$Recall = 0.9$$

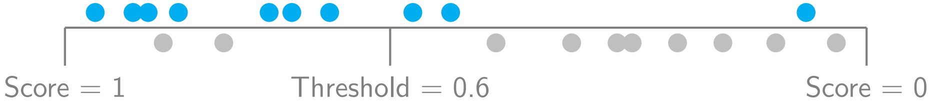
$$Specificity = 0.8$$

$$F1 = 0.857$$

Confusion Matrix - Sample Case

What if we changed the threshold?

Recall in the case of logistic regression / binary classification, we randomly picked 0.5 as the threshold. What if we picked a different threshold?

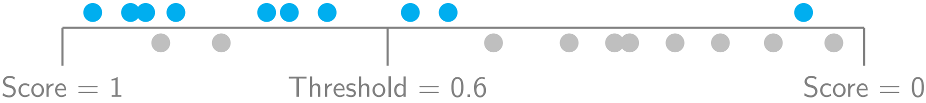


		Prediction	
		$+$ *PP*	$-$ *PN*
Actual	$+$ *P*	🟩	🟥
	$-$ *N*	🟥	🟩

Confusion Matrix - Sample Case

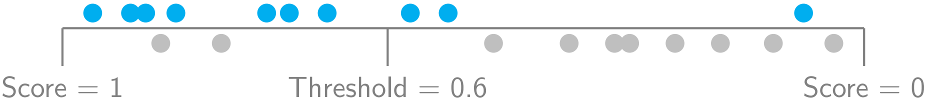
What if we changed the threshold?

Recall in the case of logistic regression / binary classification, we randomly picked 0.5 as the threshold. What if we picked a different threshold?



		Prediction	
		$+$ PP	$-$ PN
Actual	$+$ P	$TP = 7$	$FP = 2$
	$-$ N	$FN = 3$	$TN = 8$

What if we changed the threshold?



		Prediction	
		$+$ PP	$-$ PN
Actual	$+$ P	$TP = 7$	$FP = 2$
	$-$ N	$FN = 3$	$TN = 8$

$Accuracy = 0.75$

$Precision = 0.77$

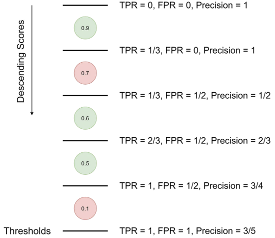
$Recall = 0.7$

$Specificity = 0.8$

$F1 = 0.733$

Plotting model performance as threshold is varied

Figures taken from Daniel Rosenberg, Towards Data Science



TPR = 0, FPR = 0, Precision = 1

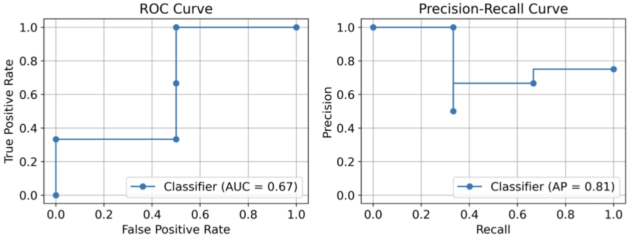
TPR = 1/3, FPR = 0, Precision = 1

TPR = 1/3, FPR = 1/2, Precision = 1/2

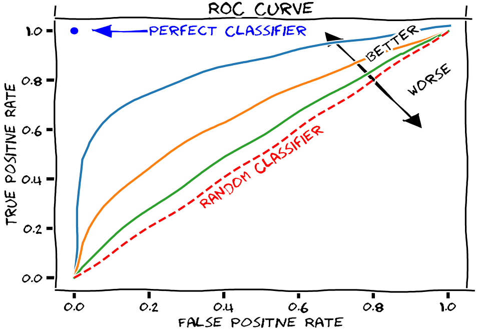
TPR = 2/3, FPR = 1/2, Precision = 2/3

TPR = 1, FPR = 1/2, Precision = 3/4

TPR = 1, FPR = 1, Precision = 3/5



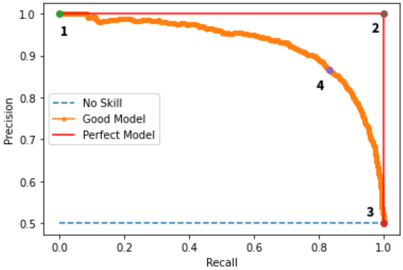
AUC - Area Under Receiver Operating Curve (ROC)



Martin Thoma, Wikipedia

- Represent model's ability to discriminate between the positive and the negative classes.
- AUC of 1.0 = perfect prediction. AUC of 0 = completely opposite prediction. AUC of 0.5 = the model is as good as random (baseline).

AUPRC - Area Under Precision Recall Curve (PRC)

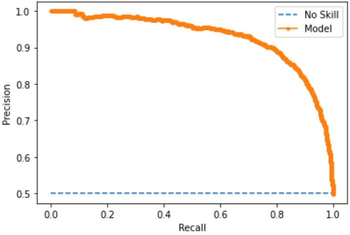


Analytics India Mag

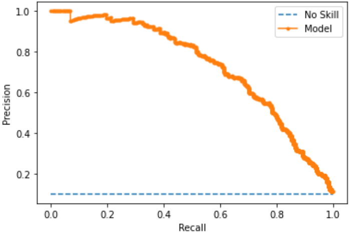
$$AUPRC = \frac{1}{m} \sum_{\hat{R} \in m} \max_{\hat{R}:\hat{R} \geq R} P(\hat{R})$$

where $P(\hat{R})$ is Precision at Recall $\hat{R}$, and m is number of thresholds. AUPRC of 1.0 = perfect prediction. AUPRC of baseline value = the model is as good as random.

AUPRC - Area Under Precision Recall Curve (PRC)



PRC of a balanced dataset

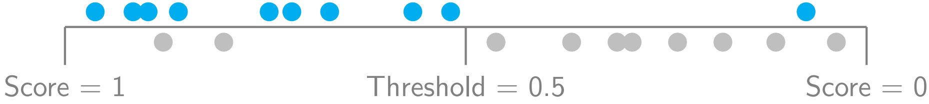


PRC of an imbalanced dataset

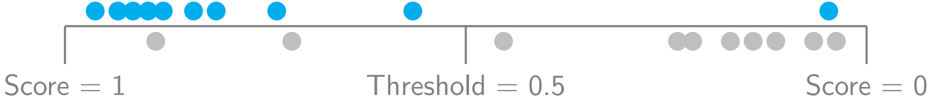
- The baseline (random chance) of the model changes depending on class imbalance, unlike the ROC in which the baseline is always at 0.5.
- PRC works better than ROC in cases of imbalanced data.
- AUPRC is also commonly known as "AP".

Which model is better?

Model A:

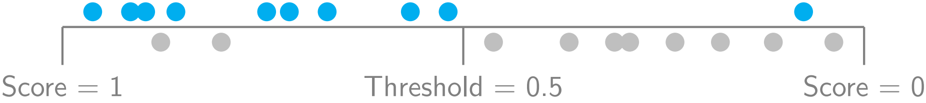


Model B:

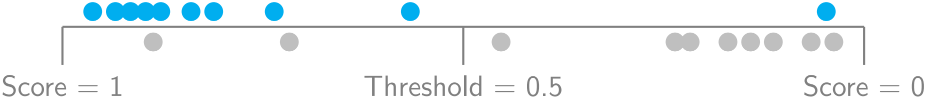


Which model is better?

Model A:



Model B:



- Both models have the same exact Confusion Matrix metrices and AUC
- Is one actually better than another, or are they the same?

Log Loss (Log Likelihood Function)

$$\ell(\theta) = -\frac{1}{n}\sum_{i=1}^{n} y^{(i)} \log h(x^{(i)}) + (1 - y^{(i)}) \log (1 - h(x^{(i)}))$$

The notation could also be written as:

$$\ell(\theta) = -\frac{1}{n}\sum_{i=1}^{n} y^{(i)} \log p(y^{(i)} + (1 - y^{(i)}) \log (1 - p(y^{(i)}))$$

where y represents the class and $p(y^{(i)})$ can be interpreted as the probability of that class

- 0 is the best loss and 1 is the worst loss.
- A ”good” loss value depends on context and problem domain, thus only makes sense when comparing log losses of multiple models.
- Might lead us to a false sense of ”confidence” in prediction.

Brier Score

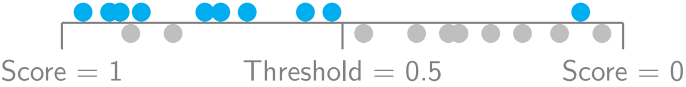
$$Brier = \frac{1}{n}\sum_{i=1}^{n}\left(p(y^{(i)}) - y^{(i)}\right)^2$$

where y represents the class and $p(y^{(i)})$ can be interpreted as the probability of that class

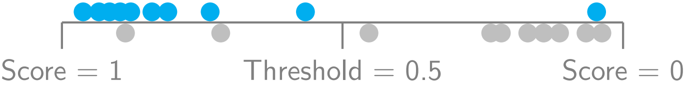
- 0 Used to check the goodness of a predicted probability score.
- More gentler than log loss in penalizing inaccurate predictions.

Log Loss vs. Brier Score

Model A:



Model B:



Log Loss

- Rewards confident correct answers and heavily penalizes confident wrong answers
- One confident wrong prediction is fatal!

Brier Score

- Gently rewards correct answers and gently penalized wrong answers

What happens during class imbalance?

- Most of the times during the training, the model would not learn on minority class examples at all.
- Using the wrong metrics might lead us to believe that the model is fine!

What happens during class imbalance?

- Accuracy: Blindly predicts majority class
- Log Loss: Majority class can dominate the loss
- AUC: Easy to keep AUC high by scoring most negatives very low
- AUPRC: More robust than AUROC
- F1 Score!

AUPRC $>$ AUC $>$ Accuracy

How to pick the right evalution metric?

Case by case basis, depending on your applications and needs:

1. When False Positives are bad. **High Precision** is a hard constraint.
   E.g. Search engine results, grammar corrections
   Metric: Recall at a certain (high) Precision → see AUPRC
2. When False Negatives are bad. **High Recall** is a hard constraint.
   E.g. Medical diagnosis
   Metric: Precision at a certain (high) Recall → see AUPRC

Multi-Class Classification

Multi-Class Classification

1. **Confusion Matrix.** *nxn*, where *n* is the number of classes.
2. All the confusion matrix related metrices (**Precision, Recall, F1 score**) can be calculated as One-vs-One or One-vs-Rest. Most practitioners nowadays use One-vs-Rest due to computationally expensive One-vs-One approach.
3. **Cohen's Kappa Score**
4. **Matthew's Correlation Coefficient**
5. **Categorial Cross Entropy.** Log loss for multi-class classification.

How to pick the right evalution metric?

1. Compare one classifier's overall performance to another in a single metric — use Matthew's correlation coefficient, Cohen's kappa, and log loss.
2. Measure a classifier's ability to differentiate between each class in balanced classification: AUC.
3. A metric that minimizes false positives and false negatives in imbalanced classification: F1 score.
4. Focus on decreasing the false positives of a single class: Precision for that class.
5. Focus on decreasing the false negatives of a single class: Recall for that class.

Regression

Regression

1. **Mean Absolute Error (MAE)**

   Penalizes all errors equally

$$MAE = \frac{1}{n} \sum_{i-1}^{n} |\hat{y}^{(i)} - y^{(i)}|$$

2. **Mean Squared Error (MSE)**

   Penalizes larger errors more

$$MAE = \frac{1}{n} \sum_{i-1}^{n} \sqrt{\left(\hat{y}^{(i)} - y^{(i)}\right)^2}$$

Task Specific Models

Task Specific Models

Object Detection Tasks

see Article by Jonathan Hui

1. **IoU**. Intersection over Union. Measures the overlap between two boundaries (true bounding box and predicted bounding box).
2. **mAP**. mean Average Precision. The average of the AP calculated for all the classes.
3. . . .

Task Specific Models

Language Tasks

1. **BLEU**. Bilingual Evaluation Understudy. This is a precision-based metric to evaluate the quality of text that are machine translated from one natural language to another.
2. **ROUGE**. Recall-Oriented Understudy for Gisting Evaluation. Used in summarization tasks to evaluate the number of words the model can recall.
3. **Perplexity**. Probabilistic measure to evaluate how confused the model is.
4. **CIDEr**. Consensus-based Image Description Evaluation. For image captioning.
5. **SPICE**. Semantic Propositional Image Caption Evaluation.
6. . . .

Benchmarks vs Real World Performance

All the Evaluation Metrics we learned today are used to compare our newly designed model with a ”baseline” model or a ”benchmark.
If the model outperforms the benchmark, then the model is good! (main points reported in papers).

In the real world, metrics are far more complex than just the benchmarks, and as ML practitioners, we often focused too much on the benchmarks at the expense of real world outcomes.

Benchmarks vs Real World Performance

In the real world, metrics are far more complex than just the benchmarks, and as ML practitioners, we often focused too much on the benchmarks at the expense of real world outcomes.

- Pick the metrics that are the most relevant for the application and/or business objectives
- Evaluate the metric end to end
- Application based metric (e.g. inference time, model size, end business/user goals, etc)
  - Model outperforms benchmarks by 10% but takes 3 minutes to spit out prediction and benchmarks took only 30 seconds.
  - Model segments tissue for surgical augmentation. Does this actually save surgeons' time? What about qualitative metrics from the surgeons?

References

Nandita Bhaskhar (2022)
CSE229 Machine Learning
Stanford University

Bex T. (2021)
Comprehensive Guide to Multiclass Classification Metrics
Towards Data Science