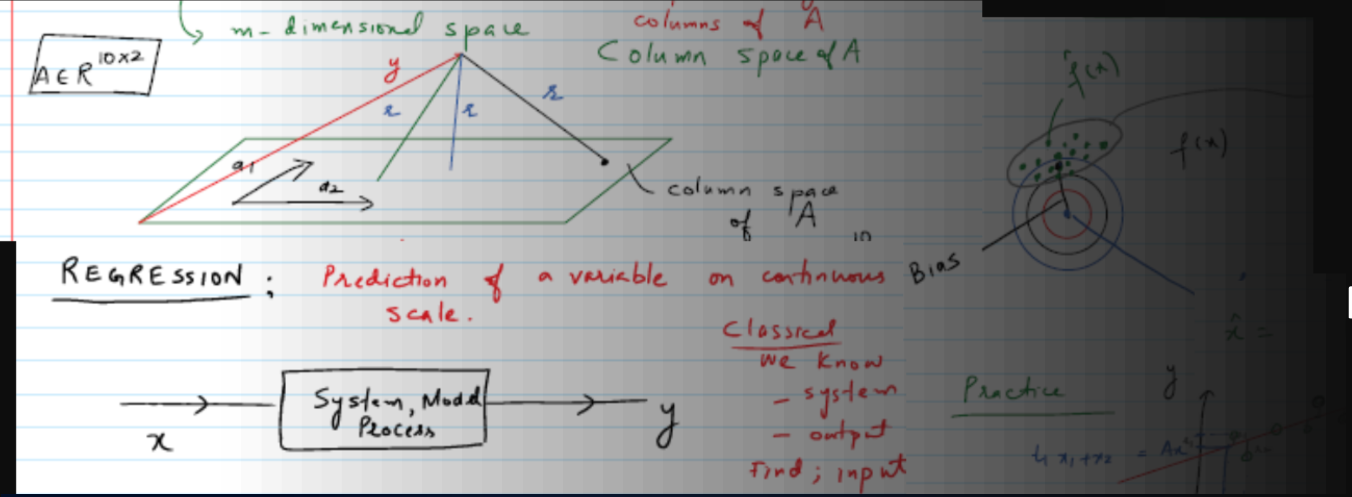


Machine Learning

EE514 – CS535

Analysis and Evaluation of Classifier's Performance and Multi-class Classification

Zubair Khalid

School of Science and Engineering
Lahore University of Management Sciences

https://www.zubairkhalid.org/ee514_2021.html

LUMS
A Not-for-Profit University

Outline

- Classification Accuracy (0/1 Loss)
- TP, TN, FP and FN
- Confusion Matrix
- Sensitivity, Specificity, Precision Trade-offs, ROC, AUC
- F1-Score and Matthew's Correlation Coefficient
- Multi-class Classification, Evaluation, Micro, Macro Averaging

Evaluation of Classification Performance

Classification Accuracy, Misclassification Rate (0/1 Loss):

$$\mathcal{L}_{0/1}(h) = \frac{1}{n}\sum_{i=1}^{n} 1 - \delta_{h(\mathbf{x_i})-y_i}$$

$$\delta_k = \begin{cases} 1, & k = 0 \\ 0 & \text{otherwise} \end{cases}$$

- For each test-point, the loss is either 0 or 1; whether the prediction is correct or incorrect.
- Averaged over n data-points, this loss is a 'Misclassification Rate'.

Interpretation:

- Misclassification Rate: Estimate of the probability that a point is incorrectly classified.
- Accuracy = 1 - Misclassification rate

Issue:

- Not meaningful when the classes are imbalanced or skewed.

Evaluation of Classification Performance

Classification Accuracy (0/1 Loss):

Example:

- Predict if a bowler will not bowl a **no-ball**?
  - Assuming 15 no-balls in an inning, a **model that says 'Yes' all the time** will have **95%** accuracy.
  - Using accuracy as performance metric, we can say that a model is very accurate, but it is not useful or valuable in fact.

Why?

- Total points: 315 (assuming other balls are legal ☺)
- No-ball label: Class 0 (4.76% are from this class)
- Not a no-ball label: Class 1 (95.24% are from this class)

Imbalanced Classes

Evaluation of Classification Performance

TP, TN, FP and FN:

- *Consider a binary classification problem.*

$$D = \{(\mathbf{x_1}, y_1), (\mathbf{x_2}, y_2), \ldots, (\mathbf{x_n}, y_n)\} \subseteq \mathcal{X}^d \times \mathcal{Y}$$

$\mathcal{Y} = \{0, 1\}$ (Referring 0 as Negative, 1 as Positive)

y - Actual labels, Ground truth, Gold labels or Standards

We have a classifier (hypothesis function) $h(\mathbf{x}) = \hat{y}$.

$y, \hat{y}$ - Positive (1) or Negative (0)

$\hat{y}$ - True if $\hat{y} = y$, False if $\hat{y} \neq y$

Evaluation of Classification Performance

TP, TN, FP and FN:

- TP - True Positive
- TN - True Negative
- FP - False Positive
- FN - False Negative

- Number of points with $y = 1$ and are classified as $\hat{y} = 1$
- Number of points with $y = 0$ and are classified as $\hat{y} = 0$
- Number of points with $y = 0$ and are classified as $\hat{y} = 1$
- Number of points with $y = 1$ and are classified as $\hat{y} = 0$

Evaluation of Classification Performance

TP, TN, FP and FN:

Example:

- *Predict if a bowler will not bowl a* ***no-ball****?*
  - *15 no-balls in an inning (Total balls: 315)*
  - *Bowl no-ball (Class 0), Bowl regular ball (Class 1)*
  - *Model(*) predicted 10 no-balls (8 correct predictions, 2 incorrect)*

- TP - True Positive
- TN - True Negative
- FP - False Positive
- FN - False Negative

- TP - 298
- TN - 8
- FP - 7
- FN - 2

** Assume you have a model that has been observing the bowlers for the last 15 years and used these observations for learning.*

Evaluation of Classification Performance

Confusion Matrix (Contingency Table):

- (TP; TN; FP; FN); usefully summarized in a table, referred to as confusion matrix:
  - the rows correspond to predicted class ($\hat{y}$)
  - and the columns to true class (y)

		Actual Labels		
Predicted Labels		**1 (Positive)**	**0 (Negative)**	Total
	1 (Positive)	**TP**	**FP**	Predicted Total Positives
	0 (Negative)	**FN**	**TN**	Predicted Total Negatives
	Total	P= TP+FN Actual Total Positives	N= P+TN Actual Total Negatives	

Evaluation of Classification Performance

Confusion Matrix:

Example:

- *Disease Detection :*

Given pathology reports and scans, predict heart disease

- *Yes: 1, No: 0*

		Actual Labels		
		1 (Positive)	**0 (Negative)**	Total
Predicted Labels	**1 (Positive)**	**TP = 100**	**FP = 10**	110
	0 (Negative)	**FN = 5**	**TN = 50**	55
	Total	P = 105	N = 60	

Interpretation:

Out of 165 cases

- *Predicted: "Yes" 110 times, and "No" 55 times*
- *In reality: "Yes" 105 times, and "No" 60 times*

Evaluation of Classification Performance

Confusion Matrix:

Example:

- *Predict if a bowler will not bowl a **no-ball**?*

		Actual Labels		
		1 (Positive)	**0 (Negative)**	Total
Predicted Labels	**1 (Positive)**	**TP = 298**	**FP = 7**	305
	0 (Negative)	**FN = 2**	**TN = 8**	10
	Total	P = 300	N = 15	

Interpretation:

Out of 315 balls, we had 15 no-balls.

- *Model predicted 305 regular balls and 10 no-balls (8 correct predictions, 2 incorrect).*

Evaluation of Classification Performance

Confusion Matrix:

		Actual Labels		
		1 (Positive)	0 (Negative)	Total
Predicted Labels	1 (Positive)	TP	FP	*Predicted Total Positives*
	0 (Negative)	FN	TN	*Predicted Total Negatives*
	Total	P= TP+FN *Actual Total Positives*	N= P+TN *Actual Total Negatives*	

Metrics using Confusion Matrix:

- ***Accuracy***: *Overall, how frequently is the classifier correct?*

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

- ***Misclassification or Error Rate***: *Overall, how frequently is it wrong?*

$$1 - \text{Accuracy} = \frac{\text{FP} + \text{FN}}{\text{Total}} = \frac{\text{FP} + \text{FN}}{\text{P} + \text{N}}$$

- ***Sensitivity or Recall or True Positive Rate (TPR)***: *How often does it predict Positive when it is actually Positive?*

$$TPR = S_e = \frac{\text{TP}}{\text{TP} + \text{FN}} = \frac{\text{TP}}{\text{P}}$$

Evaluation of Classification Performance

Confusion Matrix:

		Actual Labels		
Predicted Labels		1 (Positive)	0 (Negative)	Total
	1 (Positive)	TP	FP	*Predicted Total Positives*
	0 (Negative)	FN	TN	*Predicted Total Negatives*
	Total	P= TP+FN *Actual Total Positives*	N= P+TN *Actual Total Negatives*	

Metrics using Confusion Matrix:

- ***False Positive Rate:*** *Actual Negative, how often does it predict Positive?*

$$FPR = \frac{\mathrm{FP}}{\mathrm{TN} + \mathrm{FP}} = \frac{\mathrm{FP}}{\mathrm{N}}$$

- ***Specificity or True Negative Rate (TNR):*** *When it's actually Negative, how often does it predict Negative?*

$$TNR = S_p = \frac{\mathrm{TN}}{\mathrm{TN} + \mathrm{FP}} = \frac{\mathrm{TN}}{\mathrm{N}} = 1 - FPR$$

- ***Precision:*** *When it predicts Positive, how often is it Positive?*

$$\mathrm{Precision} = \frac{\mathrm{TP}}{\mathrm{TP} + \mathrm{FP}}$$

Evaluation of Classification Performance

Confusion Matrix Metrics:

	Actual Labels			
Predicted Labels		1 (Positive)	0 (Negative)	Total
	1 (Positive)	**TP**	**FP**	*Predicted Total Positives*
	0 (Negative)	**FN**	**TN**	*Predicted Total Negatives*
	Total	P= TP+FN *Actual Total Positives*	N= P+TN *Actual Total Negatives*	

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

$$\frac{\text{TN}}{\text{TN} + \text{FN}}$$ Negative Predicted Value

$$TPR = S_e = \frac{\text{TP}}{\text{TP} + \text{FN}} = \frac{\text{TP}}{\text{P}}$$

$$TNR = S_p = \frac{\text{TN}}{\text{TN} + \text{FP}} = \frac{\text{TN}}{\text{N}}$$

Evaluation of Classification Performance

Confusion Matrix:

		Actual Labels		
		1 (Positive)	0 (Negative)	Total
Predicted Labels	1 (Positive)	TP = 100	FP = 10	110
	0 (Negative)	FN = 5	TN = 50	55
	Total	P = 105	N = 60	

Metrics using Confusion Matrix (Example: Disease Prediction):

- ***Accuracy***: *Disease/Healthy prediction accuracy*

$$\text{Accuracy} = \frac{\text{TP} + \text{TN}}{\text{Total}} = \frac{\text{TP} + \text{TN}}{\text{P} + \text{N}}$$ **= (100+50)/165 = 0.91**

- ***Misclassification or Error Rate***: *Disease/Healthy prediction accuracy*

$$1 - \text{Accuracy} = \frac{\text{FP} + \text{FN}}{\text{Total}} = \frac{\text{FP} + \text{FN}}{\text{P} + \text{N}}$$ **= (10+5)/165 = 0.09**

- ***Sensitivity or Recall or True Positive Rate (TPR)***: *When it's positive, how often does the model detected disease?*

$$TPR = S_e = \frac{\text{TP}}{\text{TP} + \text{FN}} = \frac{\text{TP}}{\text{P}}$$ **= 100/105 = 0.95**

Evaluation of Classification Performance

Confusion Matrix:

		Actual Labels		
		1 (Positive)	0 (Negative)	Total
Predicted Labels	1 (Positive)	TP = 100	FP = 10	110
	0 (Negative)	FN = 5	TN = 50	55
	Total	P = 105	N = 60	

Metrics using Confusion Matrix (Example: Disease Prediction):

- ***False Positive Rate***: *Actually heathy, how often does it predict yes?*

$$FPR = \frac{\mathrm{FP}}{\mathrm{TN} + \mathrm{FP}} = \frac{\mathrm{FP}}{\mathrm{N}}$$ **= 10/60 = 0.17**

- ***Specificity or True Negative Rate (TNR)***: *When it's actually health, how often does it predict healthy?*

$$TNR = S_p = \frac{\mathrm{TN}}{\mathrm{TN} + \mathrm{FP}} = \frac{\mathrm{TN}}{\mathrm{N}}$$ **= 50/60 = 0.83**

- ***Precision***: *When it predicts disease, how often is it correct?*

$$\mathrm{Precision} = \frac{\mathrm{TP}}{\mathrm{TP} + \mathrm{FP}}$$ **= 100/110 = 0.91**

Evaluation of Classification Performance

Confusion Matrix:

Metrics using Confusion Matrix:

When to use which?

		Actual Labels	
		1 (Positive)	**0 (Negative)**
Predicted Labels	**1 (Positive)**	**TP**	**FP**
	0 (Negative)	**FN**	**TN**

- *Disease Detection: We do not want FN*

$$TPR = S_e = \frac{\mathrm{TP}}{\mathrm{TP} + \mathrm{FN}} = \frac{\mathrm{TP}}{\mathrm{P}}$$

- *Fraud Detection: We do not want FP*

$$TNR = S_p = \frac{\mathrm{TN}}{\mathrm{TN} + \mathrm{FP}} = \frac{\mathrm{TN}}{\mathrm{N}}$$

$$\mathrm{Precision} = \frac{\mathrm{TP}}{\mathrm{TP} + \mathrm{FP}}$$

Outline

- Classification Accuracy (0/1 Loss)
- TP, TN, FP and FN
- Confusion Matrix
- Sensitivity, Specificity, Precision Trade-offs, ROC, AUC
- F1-Score and Matthew's Correlation Coefficient
- Multi-class Classification, Evaluation, Micro, Macro Averaging

Evaluation of Classification Performance

Confusion Matrix:

Precision and Sensitivity (Recall) Trade-off:

- Disease Detection:

Sensitivity or Recall

$$TPR = S_e = \frac{\text{TP}}{\text{TP} + \text{FN}} = \frac{\text{TP}}{\text{P}}$$

Precision

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

- Recall or Sensitivity (S_e); how good we are at detecting diseased people.
- **Precision:** How many have been correctly diagnosed as unhealthy.
- If we have diagnosed everyone unhealthy, S_e**=1** (diagnose all unhealthy people correctly) but **Precision may be low** (because TN=0 that increases the value of FP).

		Actual Labels	
		1 (Positive)	0 (Negative)
Predicted Labels	1 (Positive)	TP	FP
	0 (Negative)	FN	TN

- We want high **Precision** and high S_e (=1, Ideally).
- We should combine precision and sensitivity to evaluate the performance of classifier.
  - F1-Score

LUMS
A Not-for-Profit University

Evaluation of Classification Performance

Confusion Matrix:

Sensitivity and Specificity Trade-off:

- Disease Detection:

Sensitivity or Recall

$$TPR = S_e = \frac{\mathrm{TP}}{\mathrm{TP} + \mathrm{FN}} = \frac{\mathrm{TP}}{\mathrm{P}}$$

Specificity

$$TNR = S_p = \frac{\mathrm{TN}}{\mathrm{TN} + \mathrm{FP}} = \frac{\mathrm{TN}}{\mathrm{N}}$$

- S_p and S_e; how good we are at detecting healthy and **diseased** people, respectively.

- If we have diagnosed everyone healthy, S_p=1 (diagnose all healthy people correctly) but S_e=0 (diagnose all unhealthy people incorrectly)

- Ideally: we want S_p= S_e= 1 (perfect sensitivity and specificity) but unrealistic.

Evaluation of Classification Performance

Confusion Matrix:

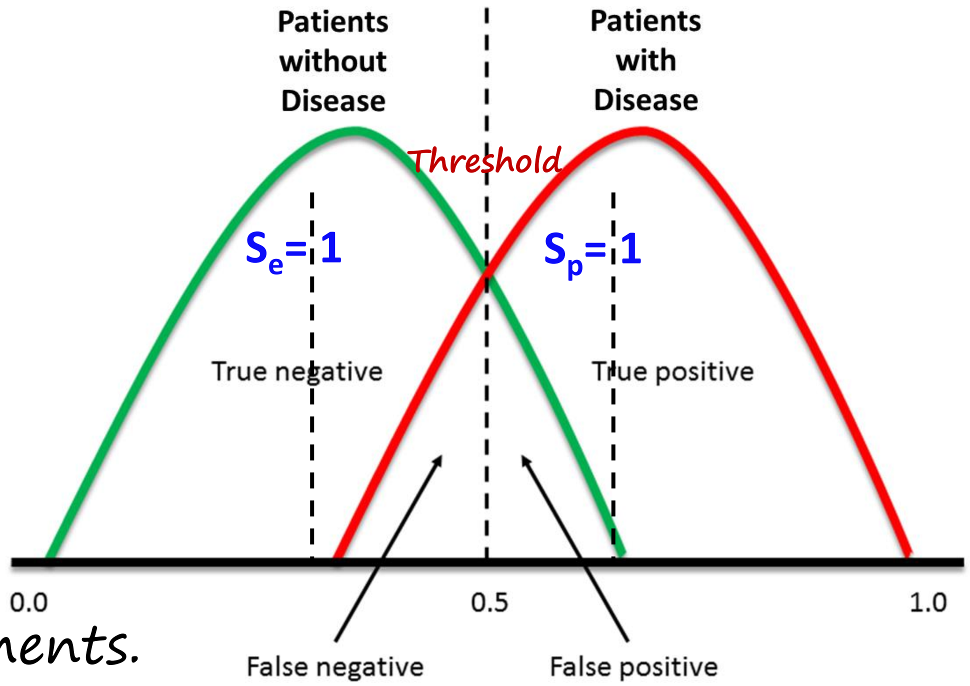
Sensitivity and Specificity Trade-off:

$$TNR = S_p = \frac{\mathrm{TN}}{\mathrm{TN}+\mathrm{FP}} = \frac{\mathrm{TN}}{\mathrm{N}}$$

$$TPR = S_e = \frac{\mathrm{TP}}{\mathrm{TP}+\mathrm{FN}} = \frac{\mathrm{TP}}{\mathrm{P}}$$

How optimal a pair of sensitivity, specificity values is?

- Is S_p= 0.8, S_e= 0.7 better than S_p= 0.7, S_e= 0.8?
- The answer depends on the application.
- In disease diagnosis;
  - happy to reduce S_p in order to increase S_e.
- In other applications, we may have different requirements.
- Trade-off is better explained by ROC curve and AUC.



Evaluation of Classification Performance

Confusion Matrix:

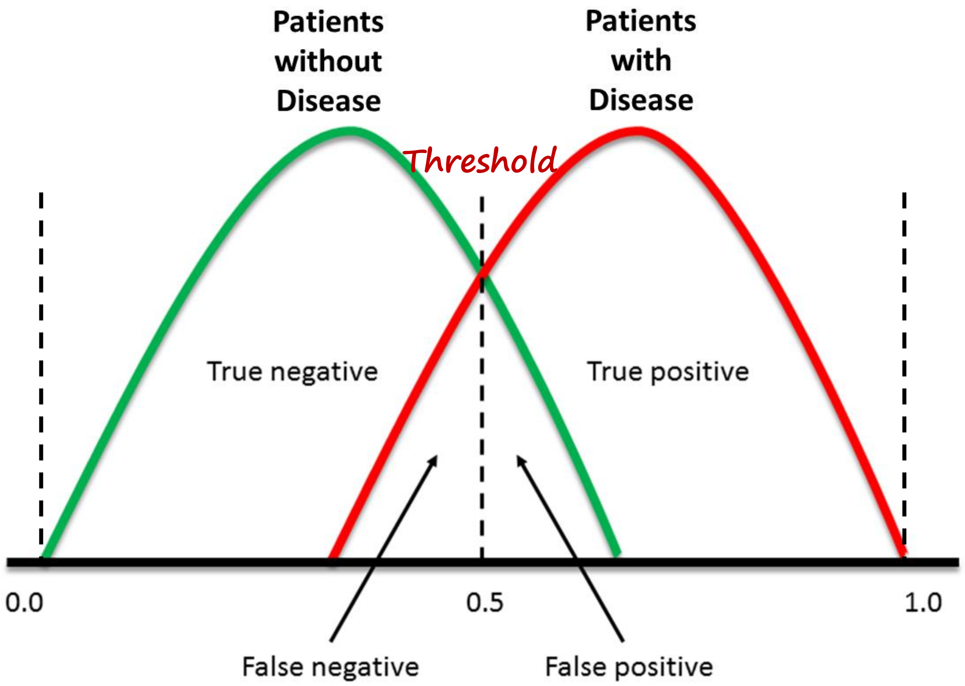
ROC (Receiver Operating Characteristic) Curve:

- Plot of TPR (Sensitivity) against FPR (1 – Specificity) for different values of threshold.

- Also referred to as Sensitivity-(1-Specificity) plot.

- Threshold of 0.0, every case is diagnosed as positive.
  - S_e= TPR = 1
  - FPR = 1
  - S_p= 0

- Threshold of 1.0, every case is diagnosed as negative.
  - S_e= TPR = 0
  - FPR = 0
  - S_p= 1



Evaluation of Classification Performance

Confusion Matrix:

ROC Curve and AUC:

- TPR (Sensitivity): how many correct positive results occur among all positive samples.

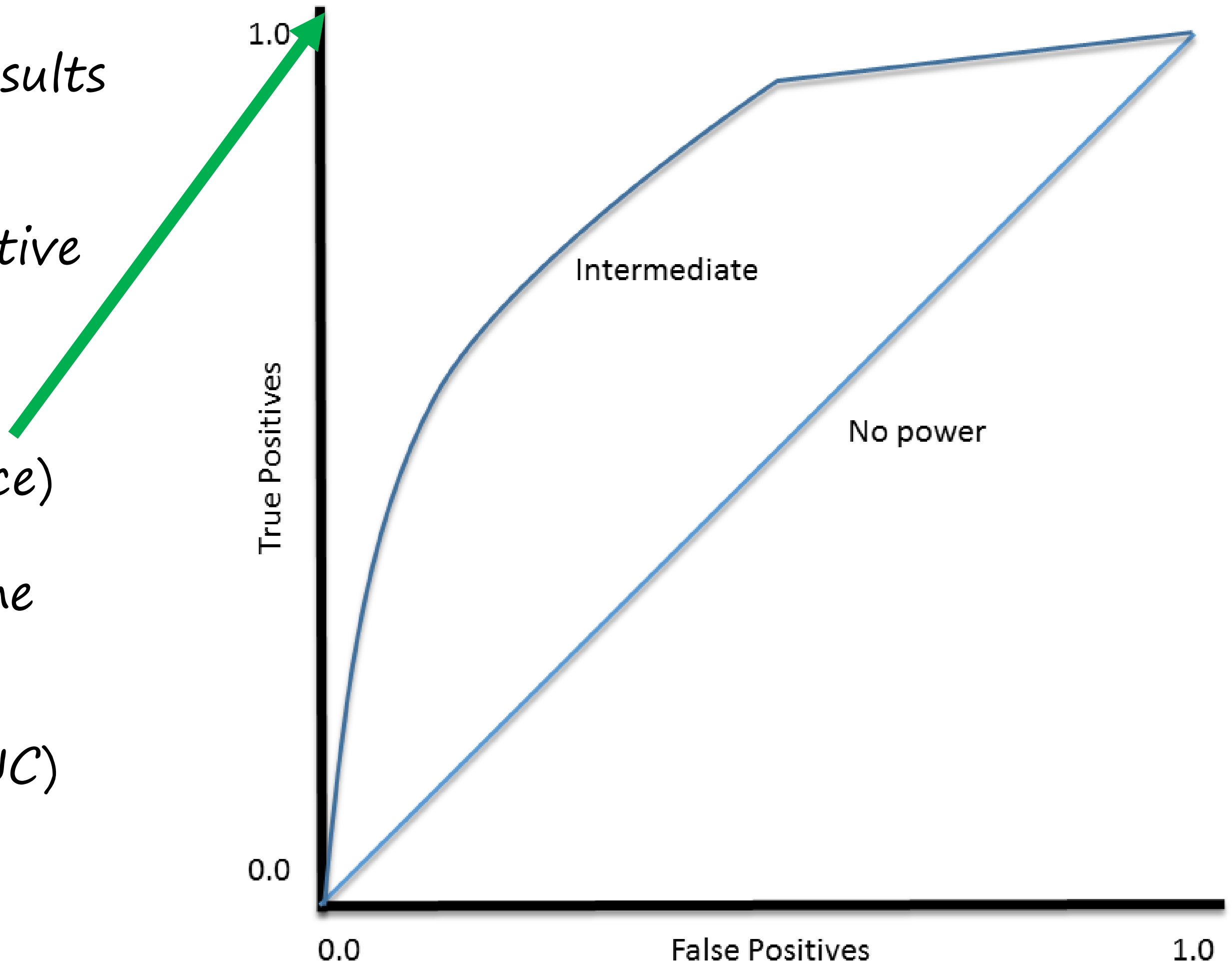
- FPR (1 – Specificity): how many incorrect positive results occur among all negative samples.

- The best possible prediction method
  - $S_e = S_p = 1$ (Upper left corner of ROC space)

- Random guess; a point along a diagonal line (the so-called line of no-discrimination), No Power!

- Area Under the ROC Curve, abbreviated as (AUC) quantifies the power of the classifier.

ROC Curve



Outline

- Classification Accuracy (0/1 Loss)
- TP, TN, FP and FN
- Confusion Matrix
- Sensitivity, Specificity, Precision Trade-offs, ROC, AUC
- F1-Score and Matthew's Correlation Coefficient
- Multi-class Classification, Evaluation, Micro, Macro Averaging

Evaluation of Classification Performance

F1-Score:

- We observed trade-off between recall and precision.

$$TPR = S_e = \frac{\mathrm{TP}}{\mathrm{TP} + \mathrm{FN}} = \frac{\mathrm{TP}}{\mathrm{P}} \qquad \mathrm{Precision} = \frac{\mathrm{TP}}{\mathrm{TP} + \mathrm{FP}}$$

- Higher levels of recall may be obtained at the price of lower values of precision.
- We need to define a single measure that combines recall and precision or other metrics to evaluate the performance of a classifier.
- Some combined measures:
  - F1 Score
  - Matthew's Correlation Coefficient
  - 11-point average precision
  - The Breakeven point

Evaluation of Classification Performance

F1 Score:

- *One measure that assesses recall and precision trade-off is weighted harmonic mean (HM) of recall and precision, that is,*

$$F_\beta = \frac{1+\beta^2}{\frac{1}{\text{Precision}} + \frac{\beta^2}{\text{Recall}}}, \quad \beta \geq 0$$

For $\beta = 1$, we have harmonic mean of precision and recall, that is,

$$F_1 = \frac{2}{\frac{1}{\text{Precision}} + \frac{1}{\text{Recall}}} = \frac{2(\text{Precision})(\text{Recall})}{(\text{Precision}) + (\text{Recall})} = \frac{2\text{TP}}{2\text{TP} + \text{FP} + \text{FN}}$$

Evaluation of Classification Performance

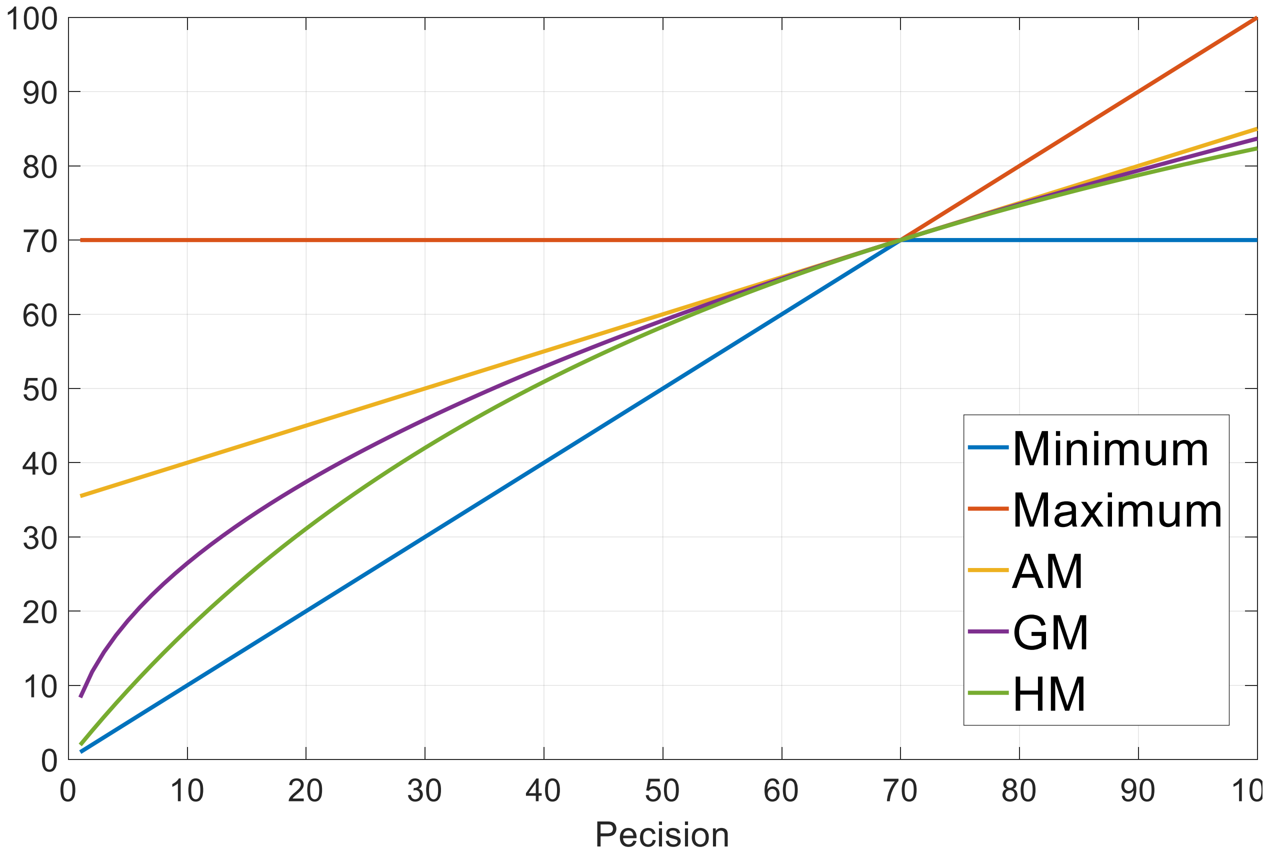
F1 Score:

Why harmonic mean?

- *We could also use arithmetic mean (AM) or geometric mean (GM).*
- *HM is preferred as it penalizes model the most; a conservative average, that is, for two real positive numbers, we have*

$$\text{HM} \leq \text{GM} \leq \text{AM}$$

- *Improvement in HM implies improvement in AM or GM.*



Different means, minimum and maximum against precision. Recall=70% is fixed.

Evaluation of Classification Performance

Matthew's Correlation Coefficient (MCC):

- Precision, Recall and F1-score are asymmetric. Get a different result if the classes are switched.
- Matthew's correlation coefficient determines the correlation between true class and predicted class. The higher the correlation between true and predicted values, the better the prediction.
- Defined as

$$\mathrm{MCC} = \frac{(\mathrm{TP})(\mathrm{TN}) - (\mathrm{FP})(\mathrm{FN})}{\sqrt{(\mathrm{TP}+\mathrm{FN})(\mathrm{TP}+\mathrm{FP})(\mathrm{TN}+\mathrm{FN})(\mathrm{TN}+\mathrm{FP})}}, \quad |\mathrm{MCC}| \leq 1$$

- MCC=1 when FP = FN = 0 (Perfect classification)
- MCC=-1 when TP = TN = 0 (Perfect misclassification)
- MCC=0; Performance of classifier is not better than a random classifier (flip coin)
- MCC is symmetric by design

Evaluation of Classification Performance

11-point Average Precision:

- Adjust threshold of the classifier such that the recall takes the following 11 values 0.0, 0.1., …, 0.9, 1.0.
- For each value of the recall, determine the precision and find the average value of precision, referred to as average precision (AP).
- This is just uniformly-spaced sampling of Precision-Recall curve and taking average value.

The Breakeven Point:

- Compute precision as a function of recall for different values of thresholds.
- When Precision = Recall, we have a breakeven.

Outline

- Classification Accuracy (0/1 Loss)
- TP, TN, FP and FN
- Confusion Matrix
- Sensitivity, Specificity, Precision Trade-offs, ROC, AUC
- F1-Score and Matthew's Correlation Coefficient
- Multi-class Classification, Evaluation, Micro, Macro Averaging

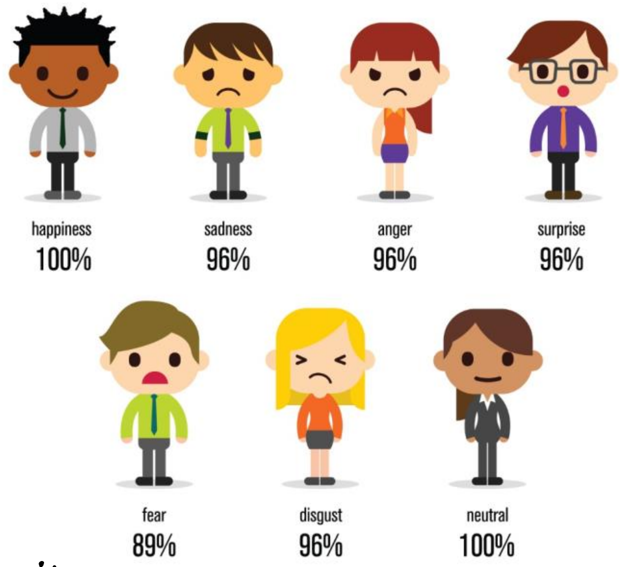
Multi-Class Classification

Formulation:

- We assume we have training data D given by

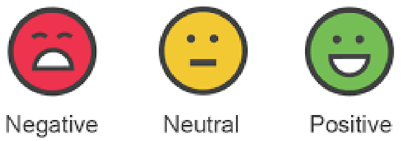
$$D = \{(\mathbf{x_1}, y_1), (\mathbf{x_2}, y_2), \ldots, (\mathbf{x_n}, y_n)\} \subseteq \mathcal{X}^d \times \mathcal{Y}$$

- $\mathcal{Y} = \{1, 2, \ldots, M\}$ (M-class classification)



Examples:

- *Emotion Detection.*
- *Vehicle Type, Make, model, color of the vehicle from the images streamed by safe city camera.*
- *Speaker Identification from Speech Signal.*
- *State (rest, ramp-up, normal, ramp-down) of the process machine in the plant.*
- *Sentiment Analysis (Categories: Positive, Negative, Neutral), Text Analysis.*
- *Take an image of the sky and determine the pollution level (healthy, moderate, hazard).*
- *Record Home WiFi signals and identify the type of appliance being operated.*



Multi-Class Classification

Implementation (Possible options using binary classifiers):

Option 1: Build a one-vs-all (OvA) one-vs-rest (OvR) classifier:

Train M different binary classifiers $h_1(\mathbf{x}), h_2(\mathbf{x}), \ldots, h_M(\mathbf{x})$.

Classifier $h_i(\mathbf{x})$ is trained to classify if $\mathbf{x}$ belongs to i-th class or not.

For a new test point $\mathbf{z}$, get scores for each classifier, that is, $s_i = h_i(\mathbf{z})$.

For example, s_i can be assigned the probability that $\mathbf{z}$ belongs to class i.

Predict the label as $\hat{y} = \max_{i=1,2,\ldots,M} \quad s_i$

Option 2: Build an all-vs-all classifier:

Train $\binom{M}{2} = \frac{(M)(M-1)}{2}$ different binary classifiers $h_{i,j}(\mathbf{x})$, $i, j = 1, 2, \ldots, k$

Classifier $h_{i,j}(\mathbf{x})$ is trained to classify if $\mathbf{x}$ belongs to i-th class or j-th class.

For a new test point $\mathbf{z}$, get scores for each classifier, that is, $s_{i,j} = h_{i,j}(\mathbf{z})$. For example, $s_{i,j} = 1$ if $\mathbf{z}$ is likely class i or 0 if $\mathbf{z}$ is likely class j.

Predict the label $\hat{y}$ that has been predicted multiple times.

There can be other options…

Evaluation of Classification Performance

Multiclass Classification:

- How do we define the measures for the evaluation of the performance of multi-class classifier?

- Macro-averaging: We compute performance for each class and then average.

- Micro-averaging: Compute confusion matrix after collecting decisions for all classes and then evaluate.

Evaluation of Classification Performance

Multiclass Classification:

Confusion Matrix

- Predict if a bowler will bowl a **no-ball, wide bowl, regular bowl?**
  - 15 no-balls, 20 wide-balls in an inning (Total balls: 335)
  - Model Predictions:

Classifier Output \ Actual	No-ball	Wide-ball	Regular ball	Precision
No-ball	8	5	20	$\frac{8}{8+5+20}$
Wide-ball	2	10	10	$\frac{10}{2+10+10}$
Regular ball	5	5	270	$\frac{270}{5+5+270}$
Recall	$\frac{8}{8+2+5}$	$\frac{10}{5+10+5}$	$\frac{270}{20+10+270}$	

Evaluation of Classification Performance

Multiclass Classification:

Confusion Matrix – Recall and Precision:

$C_{i,j}$ represents the entry of the confusion matrix at i-th row and j-th column.

Recall

- For i-th class, recall represents the fraction of data-points classified correctly, that is,

$$\text{Recall}_i = \frac{C_{i,i}}{\sum_{j=1}^{M} C_{j,i}}$$

Precision

- For i-th class, precision represents the fraction of data-points predicted to be in class i are actually in the i-th class, that is,

$$\text{Precision}_i = \frac{C_{i,i}}{\sum_{j=1}^{M} C_{i,j}}$$

	No-ball	Wide-ball	Regular ball
No-ball	8	5	20
Wide-ball	2	10	10
Regular ball	5	5	270

Accuracy

- Fraction of data points classified correctly, that is,

$$\text{Accuracy} = \frac{\sum_{i=1} C_{i,i}}{\sum_{i=1}^{M} \sum_{j=1}^{M} C_{i,j}}$$

Evaluation of Classification Performance

Multiclass Classification:

Confusion Matrix – Macro-Averaging:

- We compute performance for each class and then average.

	No-ball	Wide-ball	Regular ball
No-ball	8	5	20
Wide-ball	2	10	10
Regular ball	5	5	270

Confusion Matrix – Each Class:

Classifier Output \ Actual	No-ball	Not a No-ball
No-ball	8	25
Not a no-ball	7	295

Classifier Output \ Actual	Wide	Not Wide
Wide	10	12
Not Wide	10	303

Classifier Output \ Actual	Regular	Not Regular
Regular	270	10
Not Regular	30	25

Recall

$\frac{8}{8+7} = 0.53$ $\frac{10}{20} = 0.50$ $\frac{270}{300} = 0.90$

Macro-average Recall: $\frac{0.53+0.50+0.90}{3} = 0.64$

Evaluation of Classification Performance

Multiclass Classification:

Confusion Matrix – Micro-Averaging:

- *Compute confusion matrix after collecting decisions for all classes and then evaluate.*

	True	False
True	288	47
False	47	623

Micro-average Recall:

$$\frac{288}{335} = 0.86$$

Confusion Matrix – Each Class:

Classifier Output	Actual: No-ball	Actual: Not a No-ball
No-ball	8	25
Not a no-ball	7	295

Classifier Output	Actual: Wide	Actual: Not Wide
Wide	10	12
Not Wide	10	303

Classifier Output	Actual: Regular	Actual: Not Regular
Regular	270	10
Not Regular	30	25

Evaluation of Classification Performance

Multiclass Classification:

Micro-Averaging vs Macro Averaging:

- Note Micro-average recall= Micro-average precision = F1 Score = Accuracy (computed from confusion matrix)
  - Micro-average is termed as a global metric.
  - Consequently, it is not a good measure when classes are not balanced.

- Macro-average is relatively a better as we can see a zoomed-in picture before averaging.

- Note Macro-averaging does not take class imbalance into account.
  - **Weighted-averaging**; Similar to Macro averaging but takes a weighted mean instead where weight for each class is the total number of data-points of that class.

Weighted-average Recall: $$\frac{(15\times0.53)+(20\times0.50)+(300\times0.90)}{15+20+300} = 0.86$$

LUMS
A Not-for-Profit University

Feedback: Questions or Comments?

Email: zubair.khalid@lums.edu.pk