innovate achieve lead

Machine Learning

Dr. Monali Mavani

BITS Pilani
Pilani Campus

Disclaimer and Acknowledgement

- These content of modules & context under topics are planned by the course owner Dr. Sugata, with grateful acknowledgement to many others who made their course materials freely available online
- We here by acknowledge all the contributors for their material and inputs.
- We have provided source information wherever necessary
- Students are requested to refer to the textbook w.r.t detailed content of the presentation deck shared over canvas
- We have reduced the slides from canvas and modified the content flow to suit the requirements of the course and for ease of class presentation

Slide Source / Preparation / Review:

From BITS Pilani WILP**:** Prof.Sugata, Prof.Chetana, Prof.Rajavadhana, Prof.Monali, Prof.Sangeetha, Prof.Swarna, Prof.Pankaj

External: CS109 and CS229 Stanford lecture notes, Dr.Andrew NG and many others who made their course materials freely available online

Contents

Train_Test
70-30

- Evaluation metrics - classification

Accuracy
Precision
Recall
F1
ROC
AUC

Evaluation metrics

80% Train data = ?
or
100% Test data ⇒

Regression Evaluation

MSE
RMSE
MAP
MAE
R^2
Adj R^2

Confusion matrix

binary classification

Confusion matrix: 2 by 2 table
What happened vs What should have happened

TP => True positive.
FN = False negative.

	Pred ML system says	
actual Ground truth	Class=Yes	Class=No
Class=Yes	a (TP)	b (FN)
Class=No	c (FP)	d (TN)

Mistakes: FN and FP

Question: Which mistakes are worse than other mistakes?

law => Test.

Evaluation metric for classification problems

- **Accuracy** is the fraction of predictions our model got right. Formally, accuracy has the following definition:

$$\text{Accuracy} = \frac{\text{Number of correct predictions}}{\text{Total number of predictions}}$$

		PREDICTED CLASS	
		Class=Yes	Class=No
ACTUAL CLASS	Class=Yes	a (TP)	b (FN)
	Class=No	c (FP)	d (TN)

700

1000 → 70%

$$\text{Error rate} = \frac{\text{Number of wrong predictions}}{\text{Total number of predictions}}$$

- F n also be calculated in terms of positives and negatives as follows:

900 => 90%

$$\text{Accuracy} = \frac{a+d}{a+b+c+d} = \frac{TP+TN}{TP+TN+FP+FN}$$

Problem with Accuracy

class imbalance

	PREDICTED CLASS	
ACTUAL CLASS	Class=Yes	Class=No
Class=Yes	a (TP)	b (FN)
Class=No	c (FP)	d (TN)

- Consider a 2-class problem
  - Number of Class NO examples = 990
  - Number of Class YES examples = 10
- If a model predicts everything to be class NO, accuracy is 990/1000 = 99 %
  - This is misleading because this trivial model does not detect any class YES example
  - Detecting the rare class is usually more interesting (e.g., frauds, intrusions, defects, etc)

	PREDICTED CLASS	
ACTUAL CLASS	Class=Yes	Class=No
Class=Yes	0 TP	10 FN
Class=No	0 FP	990 TN

Which model is better?

	PREDICTED CLASS	
ACTUAL CLASS	Class=Yes	Class=No
Class=Yes	a (TP)	b (FN)
Class=No	c (FP)	d (TN)

Mileage (in kmpl)	Car Price (in cr)
9.8	High
9.12	Low
9.5	High
10	Low
….	…

Unseen Data

Mileage (in kmpl)	Car Price (in cr)
7.5	High
10	Low
….	…..

	PREDICTED	
ACTUAL	Class=Y	Class=N
Class=Y	0	10 FN
Class=N	0	990 TN

Model 1

$$\text{CarPrice} = \frac{1}{1+e^{-8.5 + 0.5\,\text{Mileage} - 1.5\,\text{Mileage}^2}}$$

Accuracy: 99%

	PREDICTED	
ACTUAL	Class=Y	Class=N
Class=Y	10 TP	0
Class=N	500 FP	490 TN

Model 2

$$\text{CarPrice} = \frac{1}{1+e^{5.5-1.5\,\text{Mileage}}}$$

Accuracy: 50%

Alternative Measures

		PREDICTED CLASS	
		Class=Yes	Class=No
ACTUAL CLASS	Class=Yes	a (TP)	b (FN)
	Class=No	c (FP)	d (TN)

Two widely used metrics used where successful detection of one of the classes is considered more significant than detection of the other classes

$$\text{Precision (p)} = \frac{a}{a+c} = \frac{TP}{TP+FP}$$

\# positives correct / # labelled positive

- What proportion of positive identifications was actually correct?
- It is the accuracy of the positive predictions
- A model that produces no false positives has a precision of 1.0.
- A system with high precision might leave some good items out but, what it returns of high quality e.g. book recommender system, safe video recommender system for kids

$$\text{Recall (r)} = \frac{a}{a+b} = \frac{TP}{TP+FN}$$

\# positives correct / # actually positive

- Also called *sensitivity* or *true positive rate*
- What proportion of actual positives was identified correctly?
- This is the ratio of positive instances that are correctly detected by the classifier
- A model that produces no false negatives has a recall of 1.0
- A system with high recall might give you lot of bad items but, it also returns most of the good items e.g robbing the shop, candidate hiring, detect shoplifters on surveillance images

Sensitivity and Specificity

- Sensitivity: the ability of a test to correctly identify patients with a disease.
- Specificity: the ability of a test to correctly identify people without the disease.

$$Recall = Sensitivity = TP\ Rate = \frac{TP}{TP + FN}$$

$$Specificity = TN\ Rate = \frac{TN}{TN + FP}$$

		ML system says	
		Class=Yes	Class=No
Ground truth	Class=Yes	a (TP)	b (FN)
	Class=No	c (FP)	d (TN)

Precision and Recall

- Precision/recall tradeoff : increasing precision reduces recall, and vice versa.
- F_1 score :
  - Suitable for class imbalance cases
  - high F_1 score if both recall and precision are High
  - F-score of 1.0, indicating perfect precision and recall

	ML system says	
	Class=Yes	Class=No
Ground truth: Class=Yes	a (TP)	b (FN)
Ground truth: Class=No	c (FP)	d (TN)

$$F_1 = \frac{2}{\frac{1}{\text{precision}} + \frac{1}{\text{recall}}} = 2 \times \frac{\text{precision} \times \text{recall}}{\text{precision} + \text{recall}} = \frac{TP}{TP + \frac{FN + FP}{2}}$$

Which Classifier is better? – Cancer prediction Low FN rate

	ML system says	
Ground truth	Class=Yes	Class=No
Class=Yes	a (TP)	b (FN)
Class=No	c (FP)	d (TN)

		PREDICTED CLASS	
		Class=Yes	Class=No
ACTUAL CLASS	Class=Yes	10 TP	0
	Class=No	10 FP	980 TN

$$\text{Precision (p)} = \frac{10}{10+10} = 0.5$$

$$\text{Recall (r)} = \frac{10}{10+0} = 1$$

$$\text{F - measure (F)} = \frac{2*1*0.5}{1+0.5} = 0.62$$

$$\text{Accuracy} = \frac{990}{1000} = 0.99$$

- FN to be minimized – use RECALL
  - Want to make sure all people with cancer will be caught
- FP to be minimized – use Precision
  - Want to say confidently that a person has cancer

		PREDICTED CLASS	
		Class=Yes	Class=No
ACTUAL CLASS	Class=Yes	1 TP	9
	Class=No	0	990

$$\text{Precision (p)} = \frac{1}{1+0} = 1$$

$$\text{Recall (r)} = \frac{1}{1+9} = 0.1$$

$$\text{F - measure (F)} = \frac{2*0.1*1}{1+0.1} = 0.18$$

$$\text{Accuracy} = \frac{991}{1000} = 0.991$$

Which Classifier is better? – SPAM email prediction Low FP rate

	ML system says	
Ground truth	Class=Yes	Class=No
Class=Yes	a (TP)	b (FN)
Class=No	c (FP)	d (TN)

- FN to be minimized – use RECALL
  - Want to make sure all SPAM emails are filtered
- FP to be minimized – use PRECISION
  - Want to confidently say email is SPAM

	PREDICTED CLASS	
ACTUAL CLASS	Class=Yes	Class=No
Class=Yes	10	0
Class=No	10	980

$$\text{Precision (p)} = \frac{10}{10+10} = 0.5$$

$$\text{Recall (r)} = \frac{10}{10+0} = 1$$

$$\text{F-measure (F)} = \frac{2*1*0.5}{1+0.5} = 0.62$$

$$\text{Accuracy} = \frac{990}{1000} = 0.99$$

	PREDICTED CLASS	
ACTUAL CLASS	Class=Yes	Class=No
Class=Yes	1	9
Class=No	0	990

$$\text{Precision (p)} = \frac{1}{1+0} = 1$$

$$\text{Recall (r)} = \frac{1}{1+9} = 0.1$$

$$\text{F-measure (F)} = \frac{2*0.1*1}{1+0.1} = 0.18$$

$$\text{Accuracy} = \frac{991}{1000} = 0.991$$

Which Classifier is better? Low Skew

Low class imbalance

	PREDICTED CLASS	
	Class=Yes	Class=No
ACTUAL CLASS Class=Yes	a (TP)	b (FN)
ACTUAL CLASS Class=No	c (FP)	d (TN)

F1 score favors classifiers that have similar precision and recall i.e. low false positives and low false negatives

T1	PREDICTED CLASS	
	Class=Yes	Class=No
ACTUAL CLASS Class=Yes	50 TP	50
ACTUAL CLASS Class=No	1	99 TN

100
100

$$\text{Precision} \quad (p) = 0.98$$
$$\text{TPR} = \text{Recall} \quad (r) = 0.5$$
$$\text{FPR} = 0.01$$

F1=0.66

T2	PREDICTED CLASS	
	Class=Yes	Class=No
ACTUAL CLASS Class=Yes	99	1
ACTUAL CLASS Class=No	10	90

$$\text{Precision} \quad (p) = 0.9$$
$$\text{TPR} = \text{Recall} \quad (r) = 0.99$$
$$\text{FPR} = 0.1$$

F1=0.94

T3	PREDICTED CLASS	
	Class=Yes	Class=No
ACTUAL CLASS Class=Yes	99	1
ACTUAL CLASS Class=No	1	99

$$\text{Precision} \quad (p) = 0.99$$
$$\text{TPR} = \text{Recall} \quad (r) = 0.99$$
$$\text{FPR} = 0.01$$

F1=0.99

Which Classifier is better? Medium Skew case

	PREDICTED CLASS	
	Class=Yes	Class=No
ACTUAL CLASS Class=Yes	a (TP)	b (FN)
ACTUAL CLASS Class=No	c (FP)	d (TN)

T1	PREDICTED CLASS	
ACTUAL CLASS	Class=Yes	Class=No
Class=Yes	50 TP	50
Class=No	10 F	990

100
1000

$$\text{Precision} \quad (p) = 0.83$$
$$\text{TPR} = \text{Recall} \quad (r) = 0.5$$
$$\text{FPR} = 0.01$$

F1=0.62

T2	PREDICTED CLASS	
ACTUAL CLASS	Class=Yes	Class=No
Class=Yes	99	1
Class=No	100 FP	900

100
1000

$$\text{Precision} \quad (p) = 0.5$$
$$\text{TPR} = \text{Recall} \quad (r) = 0.99$$
$$\text{FPR} = 0.1$$

F1=0.66

T3	PREDICTED CLASS	
ACTUAL CLASS	Class=Yes	Class=No
Class=Yes	99 TP	1
Class=No	10 FP	990

1
100
1000

$$\text{Precision} \quad (p) = 0.9$$
$$\text{TPR} = \text{Recall} \quad (r) = 0.99$$
$$\text{FPR} = 0.01$$

F1=0.94

Which Classifier is better? High Skew case

T1	PREDICTED CLASS		
		Class=Yes	Class=No
ACTUAL CLASS	Class=Yes	50	50
	Class=No	100	9900

100
1000

$$Precision\quad (p) = 0.3$$
$$TPR \quad = Recall \quad (r) = 0.5$$
$$FPR \quad = 0.01$$

F1=0.375

T2	PREDICTED CLASS		
		Class=Yes	Class=No
ACTUAL CLASS	Class=Yes	99	1
	Class=No	1000	9000

100
1000

$$Precision\quad (p) = 0.09$$
$$TPR \quad = Recall \quad (r) = 0.99$$
$$FPR \quad = 0.1$$

F1=0.165

T3	PREDICTED CLASS		
		Class=Yes	Class=No
ACTUAL CLASS	Class=Yes	99	1
	Class=No	100	9900

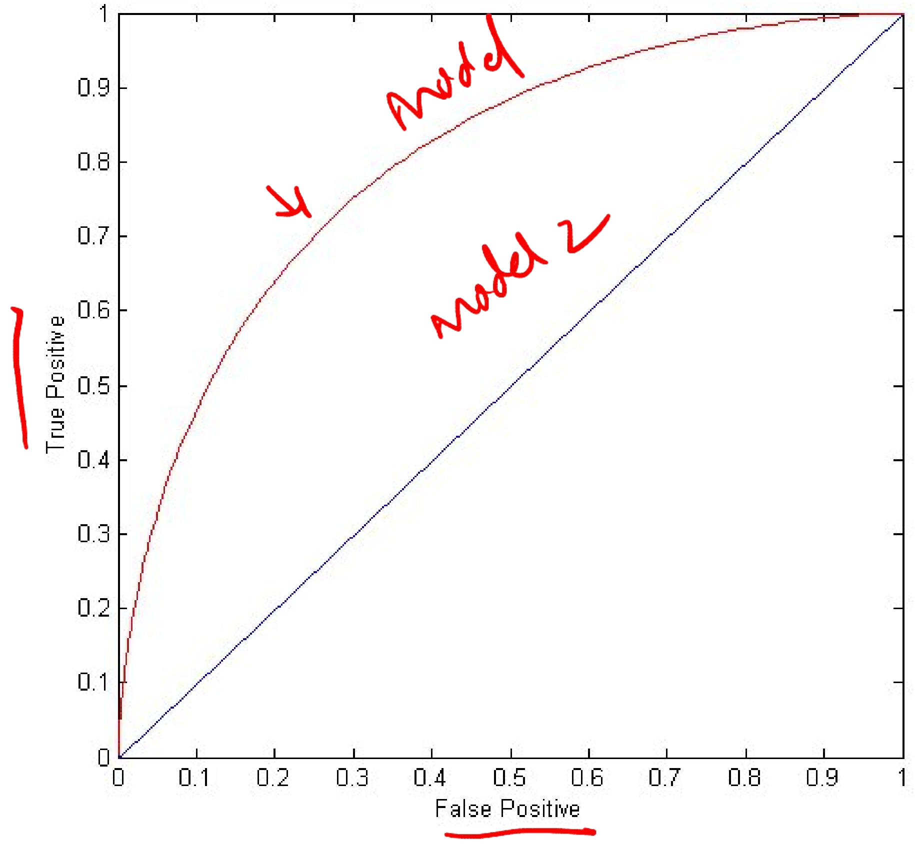
100
10000

$$Precision\quad (p) = 0.5$$
$$TPR \quad = Recall \quad (r) = 0.99$$
$$FPR \quad = 0.01$$

F1=0.66

ROC (Receiver Operating Characteristic)

- A graphical approach for displaying trade-off between detection rate and false alarm rate
- Developed in 1950s for signal detection theory to analyze noisy signals
- ROC curve plots TPR against FPR
- TPR: fraction of positive examples predicted correctly by the mode
- *FPR:* fraction of negative examples predicted as a positive class



ROC Curve

sensitivity, recall, hit rate, or true positive rate (TPR)

$$\mathrm{TPR} = \frac{\mathrm{TP}}{\mathrm{P}} = \frac{\mathrm{TP}}{\mathrm{TP} + \mathrm{FN}} = 1 - \mathrm{FNR}$$

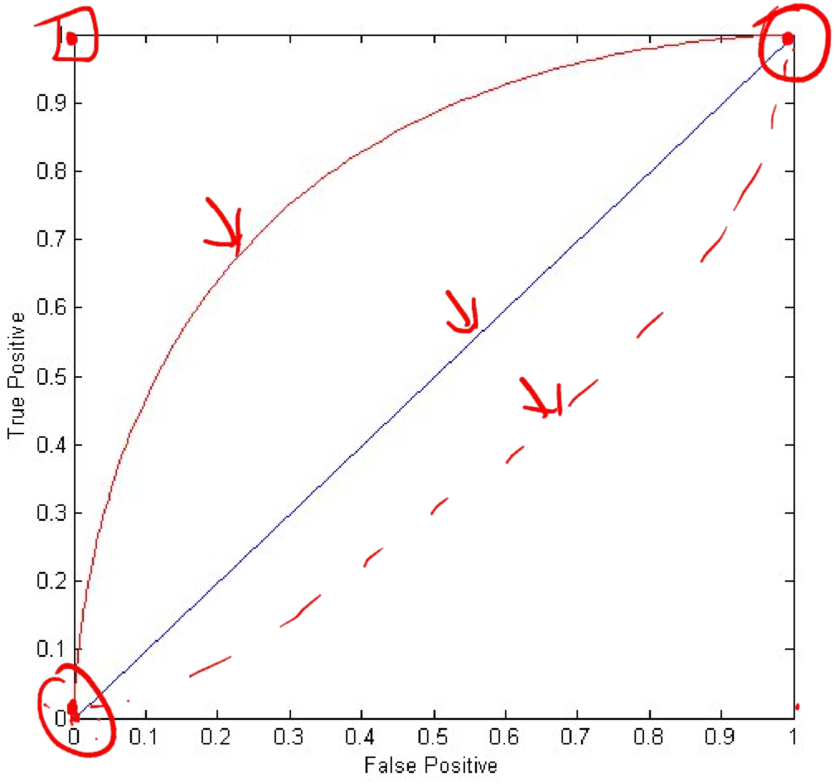
fall-out or false positive rate (FPR)

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

- Critical points along an ROC curve (TPR,FPR)
- (0,0): Model predicts every instance to be a negative class
- (1,1): Model predicts every instance to be a positive class
- (1,0): ideal

TPR (0,1)

- Diagonal line:
  - Random guessing
  - Below diagonal line:
    - prediction is opposite of the true class\
- Higher the recall (TPR), the more false positives



Comparing classifiers using AUC

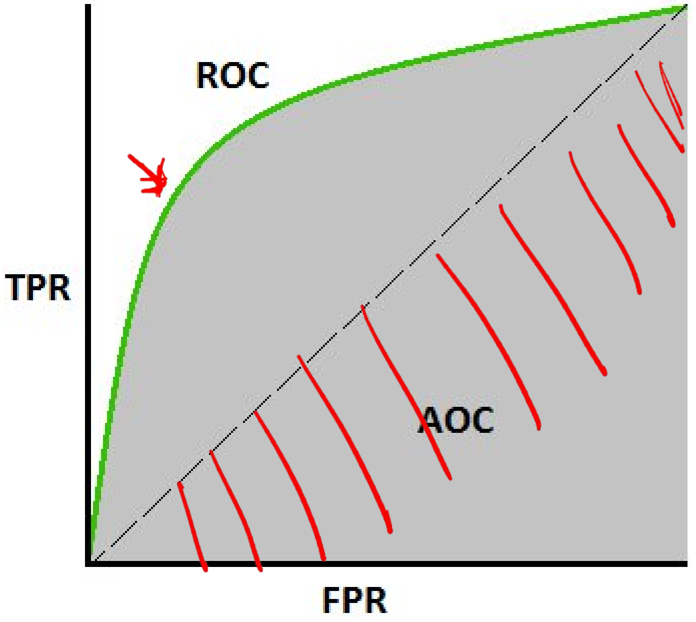
innovate achieve lead

Area Under Curve (AUC)

- Integrate Area under the curve
- Perfect score is 1
- Higher scores allow for generally better tradeoffs
- AUC of 0.5 indicates model is essentially randomly guessing
- AUC of < 0.5 indicates you're doing something wrong…
- Scikit-Learn provides a function to compute the ROC-AUC:

from sklearn.metrics import roc_auc_score

- For model comparison, AUC of ROC should be larger for the model to be superior or better performing



How to Construct an ROC curve

predicted = actual

Instance	Score	True Class
1	0.95 +	+
2	0.93 +	+
3	0.87 +	-
4	0.85 -	-
5	0.85 -	-
6	0.85	+
7	0.76	-
8	0.53	+
9	0.43	-
10	0.25	+

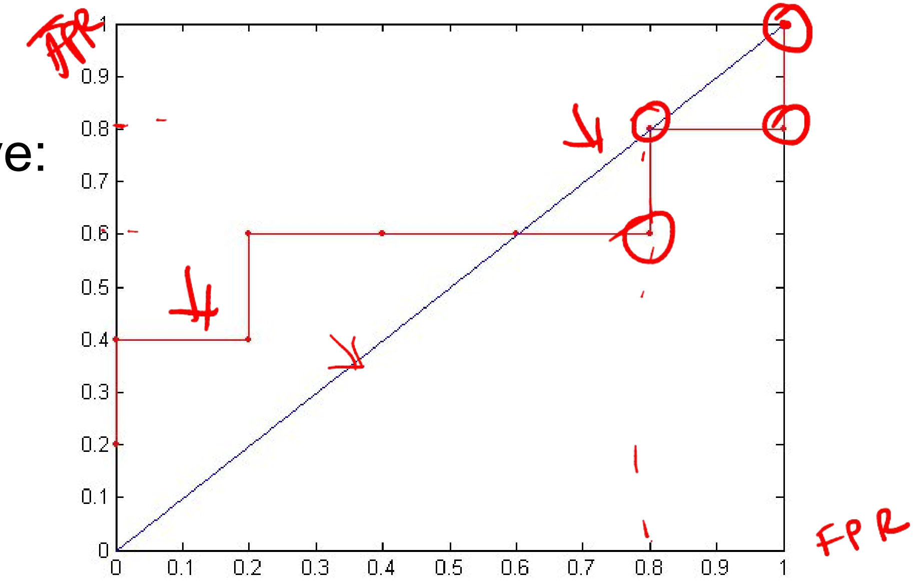
- Use a classifier that produces a continuous-valued score for each instance
  - The more likely it is for the instance to be in the + class, the higher the score
- Sort the instances in decreasing order according to the score
- Apply a threshold at each unique value of the score
- Count the number of TP, FP, TN, FN at each threshold
  - TPR = TP/(TP+FN)
  - FPR = FP/(FP + TN)

How to construct an ROC curve

Threshold
0.25
0.43
score<0.43
class=0

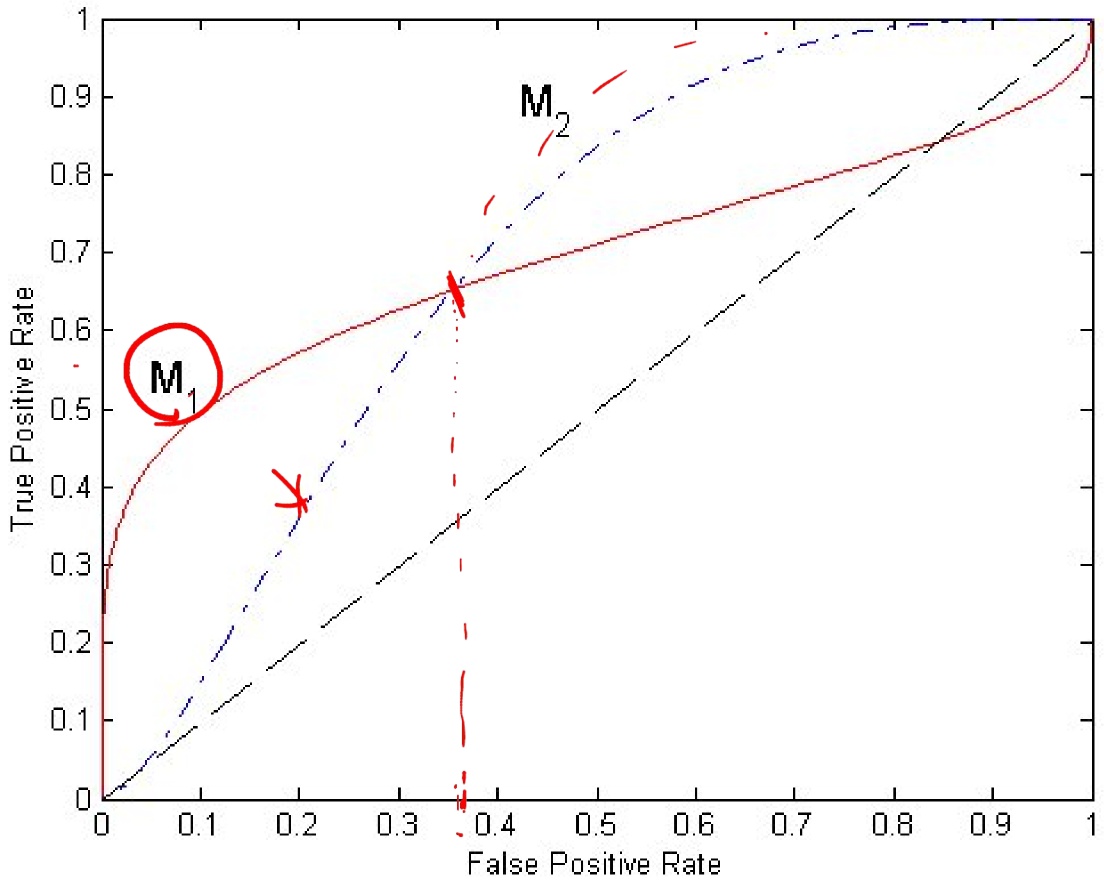
Class	+	-	+	-	-	-	+	-	+	+	
Threshold >=	0.25	0.43	0.53	0.76	0.85	0.85	0.85	0.87	0.93	0.95	1.00
TP	5	4	4	3	3	3	3	2	2	1	0
FP	5	5	4	4	3	2	1	1	0	0	0
TN	0	0	1	1	2	3	4	4	5	5	5
FN	0	1	1	2	2	2	2	3	3	4	5
TPR	1	0.8	0.8	0.6	0.6	0.6	0.6	0.4	0.4	0.2	0
FPR	1	1	0.8	0.8	0.6	0.4	0.2	0.2	0	0	0

ROC Curve:



Instance	Score	True Class
1	0.95	+
2	0.93	+
3	0.87	-
4	0.85	-
5	0.85	-
6	0.85	+
7	0.76	-
8	0.53	+
9	0.43	-
10	0.25	+

Using ROC for Model Comparison

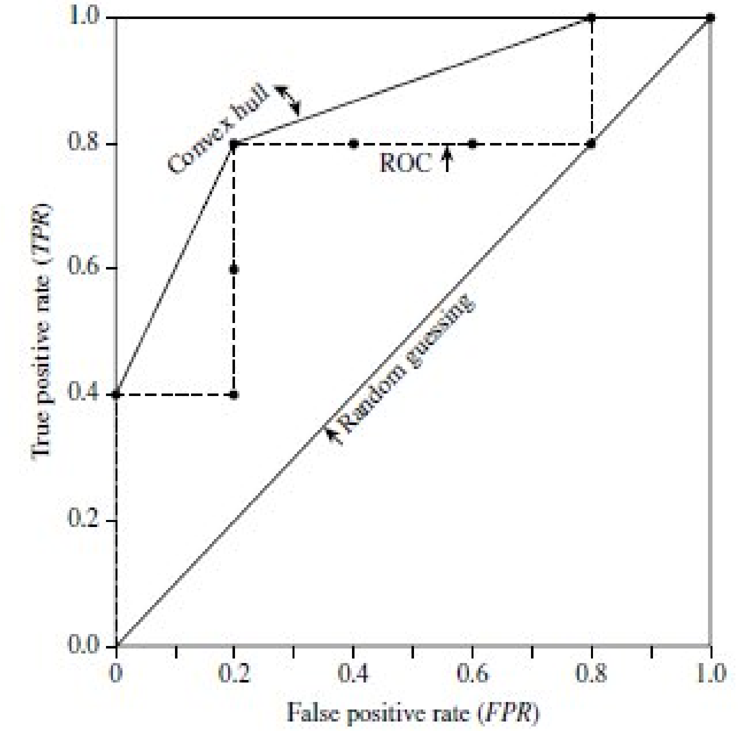


- No model consistently outperforms the other
  - $M1$ is better than $M2$ when FPR is less than 0.36
  - $M2$ is superior when FPR is greater than 0.36

Example

The table below shows the probability value (column 3) returned by a probabilistic classifier for each of the 10 tuples in a test set, sorted by decreasing probability order. The corresponding ROC is given on right hand side.

Tuple #	Class	Prob.	TP	FP	TN	FN	TPR	FPR
1	P	0.90	1	0	5	4	0.2	0
2	P	0.80	2	0	5	3	0.4	0
3	N	0.70	2	1	4	3	0.4	0.2
4	P	0.60	3	1	4	2	0.6	0.2
5	P	0.55	4	1	4	1	0.8	0.2
6	N	0.54	4	2	3	1	0.8	0.4
7	N	0.53	4	3	2	1	0.8	0.6
8	N	0.51	4	4	1	1	0.8	0.8
9	P	0.50	5	4	0	1	1.0	0.8
10	N	0.40	5	5	0	0	1.0	1.0



Solution to Class Imbalance

Down sampling →

up sampling →

- Generate Synthetic Samples
- New samples based on the distances between the point and its nearest neighbors

  E.g. Synthetic Minority Oversampling Technique, or SMOTE class in sklearn
- Change the performance metric : Use Recall, Precision or ROC curves instead of accuracy
- Try different algorithms : Some algorithms as Support Vector Machines and Tree-Based algorithms may work better with imbalanced classes.

Logistic Regression –Additional Practice Exercises

CGPA	IQ	IQ	Job Offered
5.5	6.7	100	1
5	7	105	0
8	6	90	1
9	7	105	1
6	8	120	0
7.5	7.3	110	0

Hyper parameters:
Learning Rate = 0.8
Initial Weights = (-0.1, 0.2,-0.5)
Regularization Constant = 10

For this similar problem discussed in class note that the hyper parameters are different

1. Formulate the gradient descent update equations for this problem
2. Repeat the GD for two iterations
3. Find the Loss at every iterations and interpret your observation
4. Using the results of second iteration answer below questions:
   a) Interpret the influence of the CGPA in the response variable
   b) Predict if a new candidates with IQ=5 and CGPA = 9 will be offered job or not?
5. Repeat the steps 2 to 4 by using stochastic gradient descent instead of batch gradient descent for 4 iterations. (Take any random sample from among 6 instances for these 4 iterations)

Evaluation of Classifiers–Additional Practice Exercises

Given below is a confusion matrix for medical data where the class values are yes and no for a class label attribute, cancer. Answer the following questions.

Pred

act

Classes	yes	no	Total	Recognition (%)
yes	90	210	300	30.00
no	140	9560	9700	98.56
Total	230	9770	10,000	96.40

Confusion matrix for the classes *cancer* = *yes* and *cancer* = *no*.

1. Calculate the Precision , Recall, F-Score, Error-rate, F-Score
2. Brainstorm on the use case / scenarios w.r.t given example, where precision is preferred over recall.
3. Brainstorm on the use case / scenarios w.r.t given example, where recall is preferred over precision.

References

- Ch 1,3 – Hands-on Machine Learning with Scikit-Learn, Keras, and TensorFlow by Aurélien Géron
- Ch 5 - Introduction to Data Mining by Pang-Ning Tan Michael Steinbach Vipin Kumar
- https://developers.google.com/machine-learning/crash-course/classification/roc-and-auc