National University of Singapore
School of Computing
CS3244: Machine Learning
Solution to Week 07 - Tutorial 05

Evaluation Metrics

1. **Precision, Recall, and F_1** For a given binary classification problem, a machine learning model(M) outputs a continuous score for every input sample. Table 1 shows the results for 10 samples from the model. The actual labels can be either 1(positive label) or 0 (negative label). The model output makes the final classification decision. If a threshold, p is given, the decision is made as follows.

$$label(\mathbf{x}) = [M(\mathbf{x}) \geq p)] = \begin{cases} 1 \\ 0 \end{cases}$$

Sample - $\mathbf{x}$	Model output - $M(\mathbf{x})$	Actual label ($\mathbf{y}$)
$\mathbf{x^1}$	0.435	0
$\mathbf{x^2}$	1.257	1
$\mathbf{x^3}$	2.839	1
$\mathbf{x^4}$	4.200	0
$\mathbf{x^5}$	7.432	0
$\mathbf{x^6}$	10.237	1
$\mathbf{x^7}$	12.000	1
$\mathbf{x^8}$	14.839	0
$\mathbf{x^9}$	24.207	1
$\mathbf{x^{10}}$	77.927	1

Table 1: Data information and model outputs

(a) For threshold, $p = 10$, find precision, recall and F_1 score.
Table 2 shows the decisions from the model for threshold, $p = 10$.

Sample - $\mathbf{x}$	Prediction label ($\hat{\mathbf{y}}$)	Actual label ($\mathbf{y}$)
$\mathbf{x^1}$	0	0
$\mathbf{x^2}$	0	1
$\mathbf{x^3}$	0	1
$\mathbf{x^4}$	0	0
$\mathbf{x^5}$	0	0
$\mathbf{x^6}$	1	1
$\mathbf{x^7}$	1	1
$\mathbf{x^8}$	1	0
$\mathbf{x^9}$	1	1
$\mathbf{x^{10}}$	1	1

Table 2: Data information and Model Decisions

Now we can find $TP = 4$, $FP = 1$, $TN = 3$ $FN = 2$. Precision(Pr), Recall(Re), and F_1 scores can be calculated as follows.

$$\begin{aligned} Pr &= \frac{TP}{TP + FP} \\ &= \frac{4}{4+1} = 4/5 \\ &= 0.80 \end{aligned} \tag{1}$$

$$\begin{aligned} Re &= \frac{TP}{TP + FN} \\ &= \frac{4}{4+2} = 2/3 \\ &\approx 0.67 \end{aligned} \tag{2}$$

$$\begin{aligned} F_1 &= \frac{2 \times Pr \times Re}{Pr + Re} \\ &= \frac{2 \times 4/5 \times 2/3}{4/5 + 2/3} = 8/11 \\ &\approx 0.73 \end{aligned} \tag{3}$$

(b) The number of samples is increased to m. All the model predictions($M(\mathbf{x})$) are given for m samples. Here, all the model predictions are distinct. We want to use all the model outputs as thresholds to find the best threshold for the model. Propose an optimal way to find the best threshold. Comment on the running time.
A brute force way is to calculate $F1$ scores independently using all the model outputs as thresholds. It would take $O(m^2)$ time. Let's optimize the brute force way. We can sort all the samples according to the model predictions in time $O(m \times logm)$. For the first threshold, we can find the TP, TN, FP, and FN counts to find the F_1 score. This takes $O(m)$ running time for the first threshold. We can continue this in sorted order. For the next threshold, it will take $O(1)$ time to find the F_1 score. The reason is that there will be only one change in the classification decision. Next $m-1$ thresholds would take $O(m)$ running time to find F_1 scores. The maximum can be found in $O(m)$ time. Hence, the running time to pick the best threshold is $O(m \times logm)$

(c) We have a new set of thresholds with size $q(> m)$ instead of the model predictions. What is the new running time to find the best threshold from q thresholds?
First, we can do the same procedure in part (b) on the m samples of data. The number of thresholds are increased. However, if we pick any threshold between two model outputs in the sorted order, they will create the same F_1 score. The number of different F_1 scores that can be produced from the given set of thresholds is limited to $m+1$. Hence, for every new threshold from the set of size $q(> m)$, we need to find the correct location in the sorted order of the model outputs. This can be done using binary search in $O(logm)$ time. Hence, the new running time is $O(q \times logm)$.

2. **Micro- and Macro-Averaging**

(For this question, you can try coding out the formulas and verify that the answers match!)

We have a classifier trained to predict images of cats, dogs, and pigs. The confusion matrix for the model is given below.

Predicted \ Actual	Dog	Cat	Pig
Dog	10	2	1
Cat	3	13	2
Pig	3	4	7

(a) Create the confusion matrix for each of the individual classes, dog, cat, and pig.
To generate the individual confusion matrices, we take the target class as positive examples, and all the other classes as negative examples.

Dog

Predicted \ Actual	Positive	Negative
Positive	10	3
Negative	6	26

Cat

Predicted \ Actual	Positive	Negative
Positive	13	5
Negative	6	21

Pig

Predicted \ Actual	Positive	Negative
Positive	7	7
Negative	3	28

(b) Calculate the micro-average confusion matrix, accuracy, precision, recall, and F_1 score. What do you notice about the precision, recall and F_1 score? Why is this so?
To generate the micro-average confusion matrices, we take the sum of all the individual confusion matrices.

Micro-average

Predicted \ Actual	Positive	Negative
Positive	10	5
Negative	5	25

$$\begin{aligned} Accuracy_{Micro} &= (TP + TN)/(TP + TN + FP + FN) \\ &= (10 + 25)/(10 + 25 + 5 + 5) \\ &= 0.778 \end{aligned} \tag{4}$$

$$\begin{aligned} Precision_{Micro} &= TP/(TP + FP) \\ &= 10/(10 + 5) \\ &= 0.667 \end{aligned} \tag{5}$$

$$\begin{aligned} Recall_{Micro} &= TP/(TP + FN) \\ &= 10/(10 + 5) \\ &= 0.667 \end{aligned} \tag{6}$$

$$\begin{aligned} F1_{Micro} &= ((P^{-1} + R^{-1})/2)^{-1} \\ &= ((0.667^{-1} + 0.667^{-1})/2)^{-1} \\ &= 0.667 \end{aligned} \tag{7}$$

We notice that the precision, recall, and F_1 score all has the same value. This occurs because the $FP_{Micro} = FN_{Micro}$. This is true not just for this particular case, but also generalises to all other confusion matrices. We see that for a general 3 class confusion matrix,

		Actual		
		Class 1	Class 2	Class 3
Predicted	Class 1	a	b	c
	Class 2	d	e	f
	Class 3	g	h	i

$$\begin{aligned} FP_{Micro} &= (FP_{Class_1} + FP_{Class_2} + FP_{Class_3})/3 \\ &= ((b+c) + (d+f) + (g+h))/3 \end{aligned} \tag{8}$$

$$\begin{aligned} FN_{Micro} &= (FN_{Class_1} + FN_{Class_2} + FN_{Class_3})/3 \\ &= ((d+g) + (b+h) + (c+f))/3 \end{aligned} \tag{9}$$

Rearranging the terms, we can easily see that FP_{Micro} is equal to FN_{Micro}, leading to $Precision_{Micro} = Recall_{Micro}$.

$$\begin{aligned} F1_{Micro} &= ((P^{-1} + R^{-1})/2)^{-1} \\ &= ((P^{-1} + P^{-1})/2)^{-1} \\ &= ((P^{-1})^{-1} \\ &= P \end{aligned} \tag{10}$$

Therefore, for micro-averaging of 3 classes, $Recall_{Micro} = Precision_{Micro} = F1_{Micro}$
This can be extended for confusion matrices with any number of classes.

(c) Calculate the macro-average precision and recall.
$Precision_{Dog} = 10/(10+3) = 0.769$
$Precision_{Cat} = 13/(13+5) = 0.722$
$Precision_{Pig} = 7/(7+7) = 0.5$
$Precision_{Macro} = (Precision_{Dog} + Precision_{Cat} + Precision_{Pig})/3 = 0.664$
$Recall_{Dog} = 10/(10+6) = 0.625$
$Recall_{Cat} = 13/(13+6) = 0.684$
$Recall_{Pig} = 7/(7+3) = 0.7$
$Recall_{Macro} = (Recall_{Dog} + Recall_{Cat} + Recall_{Pig})/3 = 0.670$

(d) Consider the following scenario in table 3 where there is a huge class imbalance.

Class	TP	FP
A	9	1
B	100	900
C	9	1
D	9	1

Table 3: Class imbalance Data

Calculate the $Precision_{Micro}$ and $Precision_{Macro}$ and discuss the results.

$Precision_A = Precision_C = Precision_D = 9/(9+1) = 0.9$
$Precision_B = 100/(100+900) = 0.1$
$Precision_{Macro} = (0.9+0.9+0.9+0.1)/4 = 0.7$

$TP_{Micro} = (9+100+9+9)/4 = 31.75$
$FP_{Micro} = (1+900+1+1)/4 = 225.75$
$Precision_{Micro} = TP_{Micro}/(TP_{Micro}+FP_{Micro}) = 31.75/(31.75+225.75) = 0.123$

We see that in this case, the $Precision_{Macro}$ has a much higher value. The model had a high precision on classes A, C, and D, while having a low precision on class B. $Precision_{Macro}$ takes the average of all the individual Precision values, treating each class equally. It does not consider the fact that there are a lot of examples in class B. Therefore, the $Precision_{Macro}$ is still relatively high. For $Precision_{Micro}$, the classes are not treated equally. The imbalances in classes is considered as well. Class B has a low Precision score, and makes up majority of the data as well. This leads to a very low $Precision_{Micro}$ score.