# **CVI620/DPS920- Lab 8**

# **Evaluation Measures**

| Total Mark: | 10 marks (2.5% of the total course grade)   * 7 out of 10: Blackboard submission * 3 out of 10: Lab demo |
| --- | --- |
| Submission file(s): | * Lab08.cpp * Lab08\_19W.docx including the **results** |

Please work in **groups** to complete this lab. This lab is worth 2.5% of the total course grade and will be evaluated through your written submission, as well as the lab demo. During the lab demo, group members are *randomly* selected to explain the submitted solution. Group members not present during the lab demo will lose the demo mark.

1. Add this declaration to your file:

We, ------------ (mention your names), declare that the attached assignment is our own work in accordance with the Seneca Academic Policy. We have not copied any part of this assignment, manually or electronically, from any other source including web sites, unless specified as references. We have not distributed our work to other students.

**Write a code Lab08.cpp that asks the user to input a number between 1 and 4 and runs one of the following parts:**

## **Part I: Evaluating a binary classifier**

1. Write a procedure calcPR to calculate precision and recall for a binary classifier which classifies a target as belonging to a class (true) or not (false).

int calcPR(bool \*pred, bool \*gt, int len, double \*P, double \*R);

Inputs:

pred is a pointer to classifier predictions

gt is a pointer to ground truth classes

len is the length of pred and gt arrays

Outputs:

P is a pointer to return precision

R is a pointer to return recall

Return 1 if successful, 0 otherwise.

1. Write a function to calculate Fβ:

double calcFb(double P, double R, double beta)

For example, if beta=1, the above function calculates and returns the F1 measure.

1. Write function(s) to read **alg\_bin.csv** as the classifier predictions and **gt.csv** as the ground truth. In these files, each row belongs to a data sample. The value ‘1’ means belonging to the class (true), while ‘0’ means not belonging to the class (false).
2. Write code to calculate precision, recall, and F1 for the above files. What are the values you obtain? (paste here)

***Part I: F1= 0.4000***

## **Part II: Evaluating a binary classifier with a continuous response**

Some binary classifiers do not return a certain true (1) or false (0) response. Instead, they return a probability of the sample being in the class (or similar). For example, if the response is 0.9, the classifier predicts a high probability for the sample data belonging to the class. On the other hand, if the response is 0.3, the classifier is predicting a lower probability of belonging to the class.

To convert the response of the above classifiers to a Boolean response, ***a threshold or cut-off*** value is used.

1. Write a function to convert the response of a continuous classifier (A) to a Boolean response (B) given a threshold (thresh):

void thresh\_v(double \*A, bool \*B, int len, double thresh);

If the response value is more than the threshold, convert it to true; otherwise, convert to false.

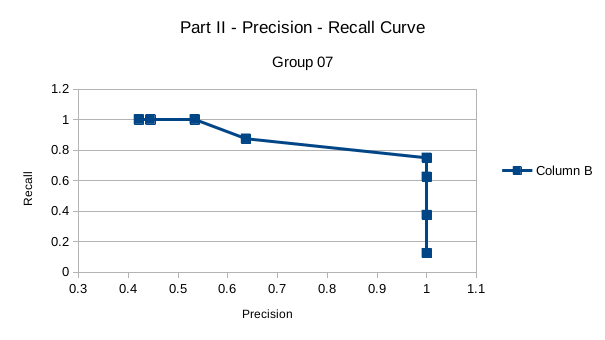
1. Write a function to read the continuous response in **alg\_dbl.csv**.
2. Using above functions (and those written in part I), write code to calculate precision, recall, and F1 for the above classifier for threshold values:

0.1, 0.2, 0.3, …, 0.8, 0.9

a) Write the evaluation measures to PR.csv (& paste the table of values here).



b) Then open PR.csv in Excel, and plot the Precision/Recall curve (& paste here).

**

c) Which threshold maximizes F1?

*Part II: max* ***F1 = 0.857143*** *at* ***threshold = 0.5***

## **Part III: Evaluating image-based classifiers**

1. Open **bitmap\_A1.png** as a grayscale image. This is the output of a classifier, where the pixel is white (255) if it is predicted to be foreground, and is black (0) if it is predicted to be background.
2. Open **bitmap\_gt.png** as ground truth.
3. Write a function that converts an image to a Boolean array. If the pixel value is more than a threshold, convert it to true; otherwise, convert to false.

void thresh\_img(Mat img, bool \*B, double thresh);

1. Using the above functions (and those in previous parts), write code to calculate the precision, recall, and F1 measure for the above classifier (and paste here).

*Part III: F1 = 0.7942*

## **Part IV: Evaluating image-based classifiers with continuous responses**

1. Open **bitmap\_A2**.png as a grayscale.
2. Similar to part II, calculate precision, recall, and F1 for the following threshold values:

25, 50, …, 225, 250

a) Write the evaluation measures to PR\_img.csv (& paste the table of values here).

b) Then open PR\_img.csv in Excel, and plot the Precision/Recall curve (& paste here).

c) Which threshold maximizes F1?

1. Specify what each member has done towards the completion of this work:

|  | Name | Task(s) |
| --- | --- | --- |
| 1 | Yathavan | Part I, II, III, IV |
| 2 | Gabby | Testing & Bug patches |
| 3 | Patrick | Testing & Bug patches |