**Precision and recall**

1. **Code for precision and recall of K-means cluster**

%% Partition Data into Two Clusters

%%

% Randomly generate the sample data.

rng default; % For reproducibility

X = [randn(100,2)\*0.75+ones(100,2);

randn(100,2)\*0.5-ones(100,2)];

figure;

plot(X(:,1),X(:,2),'.');

title 'Randomly Generated Data';

%%

% There appears to be two clusters in the data.

%%

% Partition the data into two clusters, and choose the best arrangement out of

% five initializations. Display the final output.

opts = statset('Display','final');

[idx,C] = kmeans(X,2,'Distance','cityblock',...

'Replicates',5,'Options',opts);

%%

% By default, the software initializes the replicates separately using

% \_k\_-means++.

%%

% Plot the clusters and the cluster centroids.

figure;

plot(X(idx==1,1),X(idx==1,2),'r.','MarkerSize',12)

hold on

plot(X(idx==2,1),X(idx==2,2),'b.','MarkerSize',12)

plot(C(:,1),C(:,2),'kx',...

'MarkerSize',15,'LineWidth',3)

legend('Cluster 1','Cluster 2','Centroids',...

'Location','NW')

title 'Cluster Assignments and Centroids'

hold off

%%

% You can determine how well separated the clusters are by passing |idx| to

% <docid:stats\_ug.f3984482>.

%

%% Evaluate Precision of Stop Sign Detector

% Train an ACF-based detector using pre-loaded ground truth information.

% Run the detector on the training images. Evaluate the detector and

% display the precision-recall curve.

%%

% Load the ground truth table.

load('stopSignsAndCars.mat')

stopSigns = stopSignsAndCars(:,1:2);

stopSigns.imageFilename = fullfile(toolboxdir('vision'),'visiondata', ...

stopSigns.imageFilename);

%%

% Train an ACF-based detector.

detector = trainACFObjectDetector(stopSigns,'NumStages',3);

%%

% Create a table to store the results.

numImages = height(stopSigns);

results(numImages) = struct('Boxes',[],'Scores',[]);

%%

% Run the detector on the training images. Store the results as a table.

for i = 1:numImages

I = imread(stopSigns.imageFilename{i});

[bboxes,scores] = detect(detector,I);

results(i).Boxes = bboxes;

results(i).Scores = scores;

end

results = struct2table(results);

%%

% Evaluate the results against the ground truth data. Get the precision

% statistics.

[ap,recall,precision] = evaluateDetectionPrecision(results,stopSigns(:,2));

%%

% Plot the precision-recall curve.

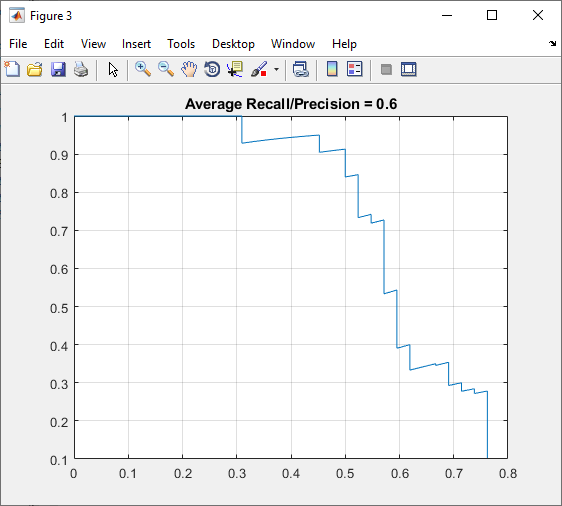
figure

plot(recall,precision)

grid on

title(sprintf('Average Recall/Precision = %.1f',ap))

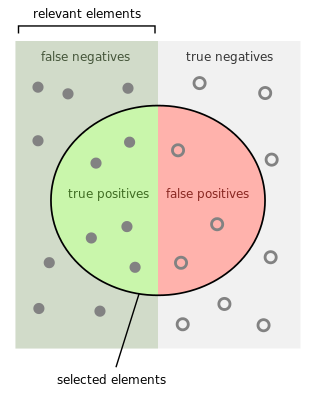
1. **Illustration of Calculating Recall And Precision**

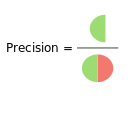
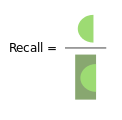


Horizon (x-axis) includes the value of recall.

Vertical (y-axis) includes the value of precision.

1. **How to calculate precision and recall**



* **Calculate the precision**
* **Calculate the recall**