

Blatt 12

Abgabe bis Mittwoch, 28. Januar 2009, 14:00 Uhr
Ausarbeitung im Sekretariat FR6052, oder bei Mikio Braun, FR6058 (notfalls unter der Tür
durchschieben),
praktischen Teil unter <https://ml01.service.tu-berlin.de/~mikio/pass.pl?conf=blatt12.conf>.

Aufgaben (ROC-Kurve und Precision-Recall-Kurve)

Implementiere die unterschiedlichen Fehlermaße und berechne daraus die ROC und Precision-Recall-Kurve.

Die Daten dazu befinden sich in dem File `sheet12_data`. Y_1 sind die vorhersagen, die man mit einer linearen Trennebene bekommt, während Y_2 mit Gauss'schen Kernen berechnet wurde. Welche der beiden Methoden liefert die besseren Vorhersagen?

1. Zähle die folgenden Arten von Vorhersagen (jeweils 2.5 Punkte):

- (a) **fp**: Anzahl der *false positives*
- (b) **tp**: Anzahl der *true positives*
- (c) **fn**: Anzahl der *false negatives*
- (d) **tn**: Anzahl der *true negatives*

Hierbei enthält das erste Argument immer die wahren Klassen und das zweite Argument die vorhergesagten Klassen.

2. Berechne die folgenden Fehlermaße (jeweils 2.5 Punkte):

- (a) **fpr**: Die *false positive rate*
- (b) **tpr**: Die *true positive rate*
- (c) **precision**: Die *precision*
- (d) **recall**: Den *recall*

3. Berechne die folgenden Fehlerkurven (jeweils 5 Punkte):

- (a) **roc_curve**: Berechne die FPR und TPR, in dem Du verschiedene Schwellwerte an die vorhergesagten Werte anlegst.
- (b) **pr_curve**: Berechne Precision und Recall wieder für verschiedene geeignet gewählte Schwellwerte.

Beachte, dass alle Funktionen bis auf `roc_curve` und `pr_curve` Vektoren mit Werten ± 1 erwarten, während diese beiden Funktionen reelwertige Vorhersagen erwarten.

```
function sheet12
% ROC-curves

sheet12_data % loads X, Y, Y1 and Y2

% plot the data
figure(1)
plot(X(Y == 1, 1), X(Y == 1, 2), 'r+', ...
      X(Y == -1, 1), X(Y == -1, 2), 'bo');
legend('positive', 'negative');

% plot the first Ys
figure(2)
```

```

scatter(X(:, 1), X(:, 2), 20, Y1, 'filled');
title('predicitons 1')

figure(3)
scatter(X(:, 1), X(:, 2), 20, Y2, 'filled');
title('predicitons 2')

figure(4)
[FPR1, TPR1] = roc_curve(Y, Y1);
[FPR2, TPR2] = roc_curve(Y, Y2);
title('ROC')
plot(FPR1, TPR1, 'r-', FPR2, TPR2, 'b-')
legend('1', '2');
xlabel('false positive rate')
ylabel('true positive rate')

figure(5)
[P1, R1] = pr_curve(Y, Y1);
[P2, R2] = pr_curve(Y, Y2);
title('ROC')
plot(P1, R1, 'r-', P2, R2, 'b-')
legend('1', '2');
xlabel('precision');
ylabel('recall');

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%
% Your solutions below

% 1a. count the number of false positives. Yt are the true labels, Yp are the
% predicted ones.
function N = fp(Yt, Yp)
% ...

% 1b. count the number of true positives.
function N = tp(Yt, Yp)
% ...

% 1c. count the number of false negatives.
function N = fn(Yt, Yp)
% ...

% 1d. count the number of true negatives.
function N = tn(Yt, Yp)
% ...

% 2a. compute the false positive rate between Yt and Yp, where Yt are the
% true labels and Yp are the predicted labels.
function R = fpr(Yt, Yp)
% ...

% 2b. compute the true positive rate
function R = tpr(Yt, Yp)
% ...

% 2c. compute the precision
function R = precision(Yt, Yp)
% ...

% 2d. compute the recall
function R = recall(Yt, Yp)

```

```
% ...  
  
% 3a. Compute the values for the roc-curve  
function [FPR, TPR] = roc_curve(Yt, Yp)  
% ...  
  
% 3b. compute the values for the precision-recall curve  
function [P, R] = pr_curve(Yt, Yp)  
% ...
```
