

LAPI – Laboratorul de Analiză și Prelucrarea Imaginilor

Facultatea de Electronică, Telecomunicații și Tehnologia Informației

Universitatea POLITEHNICA din București

Tehnici de analiză și clasificare automată a informației

Conf. dr. ing. Bogdan IONESCU
<http://imag.pub.ro/~bionescu>

București, 2015

Plan Curs

- M1. Introducere (concept, aplicații)
- M2. Prelucrarea și reprezentarea datelor de intrare
- M3. Tehnici de clasificare ne-supervizată ("clustering")
- M4. Tehnici de clasificare supervizată ("classification")
- M5. Evaluarea performanței clasificatorilor

Tehnici de analiză și clasificare automată a informației, Conf. Bogdan IONESCU 2

> M5. Evaluarea performanței clasificatorilor

- 5.1. [Introducere]
- 5.2. [Măsurile de performanță]
- 5.3. [Evaluarea performanței]
- 5.4. [Exemple de sisteme de clasificare]

Tehnici de analiză și clasificare automată a informației, Conf. Bogdan IONESCU 3

Evaluarea performanței unui sistem - principiu

Având la dispoziție un sistem de clasificare și un set de parametri de intrare, cum putem evalua performanța acestuia?

date de intrare

clasificator 1

clasa 1

clasa 2

clasa 3

Tehnici de analiză și clasificare automată a informației, Conf. Bogdan IONESCU 4

Evaluarea performanței unui sistem – principiu (cont.)

date de intrare

clasificator 2

clasa 1

clasa 2

clasa 3

Tehnici de analiză și clasificare automată a informației, Conf. Bogdan IONESCU 5

Evaluarea performanței unui sistem – principiu (cont.)

date de intrare

clasificator n

clasa 1

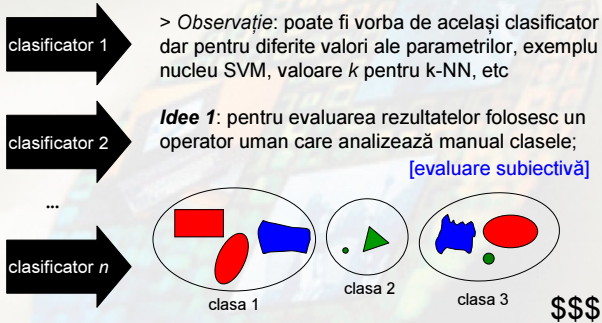
clasa 2

clasa 3

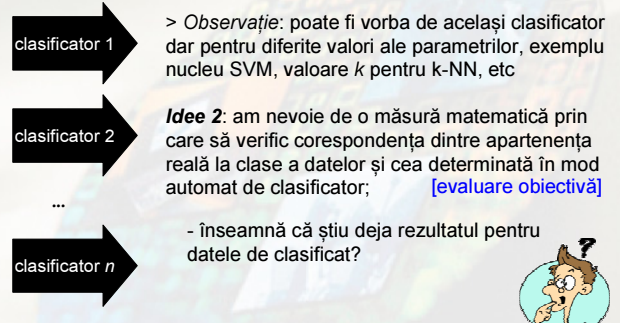
clasa 4

Tehnici de analiză și clasificare automată a informației, Conf. Bogdan IONESCU 6

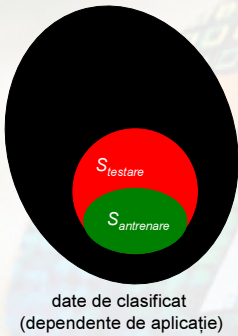
Evaluarea performanței unui sistem – principiu (cont.)



Evaluarea performanței unui sistem – principiu (cont.)

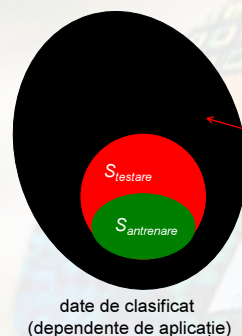


Evaluarea performanței unui sistem – principiu (cont.)



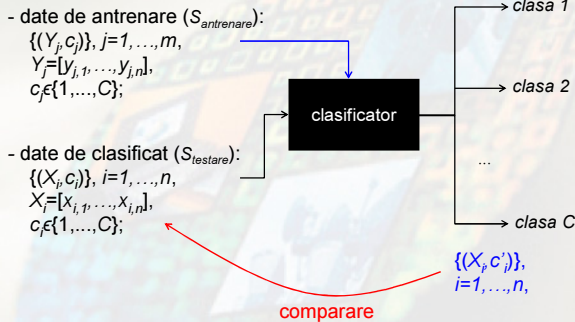
- determin un subset de date (S) pentru care cunosc/determin apartenența la clase (*ground truth*)
> suficient de mare cât să fie reprezentativ pentru restul datelor;
- determin un subset $S_{antrenare}$ pe care voi antrena clasificatorul;
- clasificatorul este testat pe $S - S_{antrenare} = S_{testare}$ (cunosc *ground truth*);

Evaluarea performanței unui sistem – principiu (cont.)



- clasificatorul este **optimizat** (alegere parametri) folosind $S_{antrenare}$ pentru antrenare și verificare performanță pe $S_{testare}$;
- odată clasificatorul optimizat acesta este aplicat datelor necunoscute de clasificat (sperând cel puțin să mențină performanța obținută pe setul S);
- cum evaluăm performanța la pasul 3 & 4? **calculul unor erori**;

Măsuri de performanță



Măsuri de performanță (cont.)

> să considerăm cazul unui clasificator binar, $c_i \in \{1, 2\}$;

TP, FP, TN, FN

		realitate (ground truth)	
		c_1^*	c_2
rezultat în urma clasificării	c_1	TP	
	c_2		

TP – True Positive, clasificare corectă, în realitate data este în c_1 , iar în urma clasificării am obținut aceeași clasă;

*clasa principală vizată de clasificator (ex. da vs. nu).

Măsuri de performanță (cont.)

> să considerăm cazul unu clasificator binar, $c_i \in \{1,2\}$ (cont.);

TP, FP, TN, FN (cont.)

		realitate (ground truth)	
		c_1^*	c_2
rezultat în urma clasificării	c_1	TP	FP
	c_2		

FP – False Positive, clasificare falsă, în realitate data este în c_2 iar în urma clasificării am obținut că ar fi în c_1 ;

*clasa principală vizată de clasificator (ex. da vs. nu).

Măsuri de performanță (cont.)

> să considerăm cazul unu clasificator binar, $c_i \in \{1,2\}$ (cont.);

TP, FP, TN, FN (cont.)

		realitate (ground truth)	
		c_1^*	c_2
rezultat în urma clasificării	c_1	TP	FP
	c_2	FN	

FN – False Negative, non detecție, în realitate data este în c_1 iar în urma clasificării am obținut că ar fi în c_2 ;

*clasa principală vizată de clasificator (ex. da vs. nu).

Măsuri de performanță (cont.)

> să considerăm cazul unu clasificator binar, $c_i \in \{1,2\}$ (cont.);

TP, FP, TN, FN (cont.)

		realitate (ground truth)	
		c_1^*	c_2
rezultat în urma clasificării	c_1	TP	FP
	c_2	FN	TN

TN – True Negative, clasificare corectă pentru clasa opusă, în realitate data este în c_2 iar în urma clasificării obținem tot c_2 ;

*clasa principală vizată de clasificator (ex. da vs. nu).

Măsuri de performanță (cont.)

> să considerăm cazul unu clasificator binar, $c_i \in \{1,2\}$ (cont.);

Precision/Recall

		realitate (ground truth)	
		c_1^*	c_2
rezultat în urma clasificării	c_1	TP	FP
	c_2	FN	TN

$$Precision = \frac{TP}{TP + FP} \quad \begin{array}{l} \text{- măsură a falselor clasificări;} \\ \text{- } FP=0 \text{ rezultă } 100\%. \end{array}$$

*clasa principală vizată de clasificator (ex. da vs. nu).

Măsuri de performanță (cont.)

> să considerăm cazul unu clasificator binar, $c_i \in \{1,2\}$ (cont.);

Precision/Recall (cont.)

		realitate (ground truth)	
		c_1^*	c_2
rezultat în urma clasificării	c_1	TP	FP
	c_2	FN	TN

$$Recall = \frac{TP}{TP + FN} \quad \begin{array}{l} \text{- măsură a non-detețiilor;} \\ \text{- } FN=0 \text{ rezultă } 100\%. \end{array}$$

*clasa principală vizată de clasificator (ex. da vs. nu).

Măsuri de performanță (cont.)

> să considerăm cazul unu clasificator binar, $c_i \in \{1,2\}$ (cont.);

Precision/Recall (cont.)

$$Precision = \frac{TP}{TP + FP}, \quad Recall = \frac{TP}{TP + FN}$$

- exemplu numeric #1 ($c_1 = \text{film}$, $c_2 = \text{nimic}$);

nr.	vreme	vizită părinți	buget	decizie	ground truth
#1	soare	da	bogat	film	film
#2	soare	nu	bogat	nimic	nimic
#3	vânt	da	bogat	nimic	film
#4	ploaie	nu	bogat	film	nimic
#5	ploaie	da	sărac	film	film
#6	vânt	nu	sărac	film	film
#7	vânt	nu	bogat	film	nimic

TP = 3 , **FP = 2** , **FN = 1** , **Precision = 60%** , **Recall = 75%**

Măsuri de performanță (cont.)

> să considerăm cazul unu clasificator binar, $c_i \in \{1,2\}$ (cont.);

$$\text{Precision/Recall (cont.)} \quad \text{Precision} = \frac{TP}{TP + FP}, \text{Recall} = \frac{TP}{TP + FN}$$

- exemplu numeric #2 ($c_1 = \text{film}$, $c_2 = \text{nimic}$);

nr.	vreme	vizită părinți	buget	decizie	ground truth
#1	soare	da	bogat	film	film
#2	soare	nu	bogat	film	nimic
#3	vânt	da	bogat	film	film
#4	ploaie	nu	bogat	film	nimic
#5	ploaie	da	sărac	film	film
#6	vânt	nu	sărac	film	film
#7	vânt	nu	bogat	film	nimic

$TP = 4$, $FP = 3$, $FN = 0$, $Precision = 57\%$, $Recall = 100\%$

Măsuri de performanță (cont.)

> să considerăm cazul unu clasificator binar, $c_i \in \{1,2\}$ (cont.);

$$\text{Precision/Recall (cont.)} \quad \text{Precision} = \frac{TP}{TP + FP}, \text{Recall} = \frac{TP}{TP + FN}$$

- ce este mai important, precision sau recall?

- ce este mai important să obținem, cât mai puține clasificări false sau cât mai puține non-detectii?

- depinde de aplicație!

• **web**: ex. sistem de căutare a informației; cât de important este să găsim toate datele de un anumit tip de pe tot Internet-ul?

• **forensics**: ex. sistem de căutare a unei persoane pe baza profilului; cât de important este să găsim toate persoanele care corespund profilului căutat?

Măsuri de performanță (cont.)

> să considerăm cazul unu clasificator binar, $c_i \in \{1,2\}$ (cont.);

F-measure

- există o măsură care combină *precision* și *recall* într-un mod unitar:

$$F\text{-measure} = (1 + \beta^2) \frac{Precision \cdot Recall}{\beta^2 \cdot Precision + Recall}$$

unde β este o constantă:

$$\beta = 1 \Rightarrow F1 = 2 \frac{Precision \cdot Recall}{Precision + Recall} \quad [\text{contribuție egală, medie armonică}]$$

$$\beta = 2 \Rightarrow F2 = 5 \frac{Precision \cdot Recall}{4 \cdot Precision + Recall} \quad [\text{pondere mai mare Recall}]$$

Măsuri de performanță (cont.)

> să considerăm cazul unu clasificator binar, $c_i \in \{1,2\}$ (cont.);

Accuracy

- o măsură a numărului de clasificări corecte:

		realitate (ground truth)	
		c_1^*	c_2
rezultat în urma clasificării	c_1	TP	FP
	c_2	FN	TN

$$Accuracy = \frac{TP + TN}{TP + FP + FN + TN} \quad - TP + FP + FN + TN = \text{numărul total de date};$$

Măsuri de performanță (cont.)

> să considerăm cazul unu clasificator binar, $c_i \in \{1,2\}$ (cont.);

Mean Average Precision (MAP)

- o măsură a *preciziei medii*; spre deosebire de *Precision* și *Recall*, MAP ține cont de ordinea în care sunt clasate datele (mai important să avem rezultate corecte în primele rezultate);

- o metrică adaptată unui scenariu de tip "information retrieval" în care rezultatele sunt ordonate în ordinea descrescătoare a asemănării cu datele căutate (echivalent clasă);

- cum poate fi adaptată pentru problema clasificării?

[reprezentăm datele clasificate în ordinea descrescătoare a măsurii de încredere ("confidence level") furnizată de clasificator, astfel obținem o ordonare a acestora]

Măsuri de performanță (cont.)

> să considerăm cazul unu clasificator binar, $c_i \in \{1,2\}$ (cont.);

Mean Average Precision (MAP; cont.)

$$AP = \frac{\sum_{k=1}^n P(k) \cdot rel(k)}{\#rel}$$

unde $\#rel$ reprezintă numărul de date relevante existente în clasa curentă, n reprezintă numărul de date de clasificat, $P(k)$ reprezintă *Precision* calculat pentru primele k date, $rel(k) = 1$ dacă data de pe poziția k este relevantă pentru clasă și 0 altfel;

$$MAP = \frac{1}{Q} \sum_{q=1}^Q AP(q)$$

unde q reprezintă clasele (adoptat din notație inițială unde reprezenta "query") iar Q este numărul de clase în care clasificăm datele.

Măsuri de performanță (cont.)

> să considerăm cazul general, multiclassă, $c_i \in \{1, \dots, C\}$; cum se aplică măsurile de performanță definite anterior?

> sunt calculate pentru fiecare clasă vizată în parte, "one-vs-all".

Confusion Matrix

rezultat în urma clasificării

	c_1	c_2	...	c_C
c_1	5	2	...	1
c_2	0	6	...	0
...				
c_C				

câte date care erau în realitate în clasa c_2 au fost clasificate de fapt în c_1 ;

Măsuri de performanță (cont.)

> să considerăm cazul general, multiclassă, $c_i \in \{1, \dots, C\}$ (cont.);

Confusion Matrix (cont.)

rezultat în urma clasificării

	c_1	c_2	...	c_C
c_1	5	2	...	1
c_2	0	6	...	0
...				
c_C				

câte date care erau în realitate în clasa c_2 au fost clasificate de fapt în c_2 ;

> cum arată matricea de confuzie pentru un sistem de clasificare perfect? **diagonală**

Măsuri de performanță (cont.)

> să considerăm cazul general, multiclassă, $c_i \in \{1, \dots, C\}$ (cont.);

Confusion Matrix (cont.)

c_2 : $TP = 6$

rezultat în urma clasificării

	c_1	c_2	c_i	c_C
c_1	5	2	3	1
c_2	0	6	1	0
c_i	1	2	11	3
c_C	1	10	0	3

> putem pe baza matricei de confuzie să estimăm valorile TP , FP , TN , FN (și astfel $Precision/Recall$)? **să luăm exemplul pe c_2**

Măsuri de performanță (cont.)

> să considerăm cazul general, multiclassă, $c_i \in \{1, \dots, C\}$ (cont.);

Confusion Matrix (cont.)

c_2 : $TP = 6$, $FP = 14$

rezultat în urma clasificării

	c_1	c_2	c_i	c_C
c_1	5	2	3	1
c_2	0	6	1	0
c_i	1	2	11	3
c_C	1	10	0	3

> putem pe baza matricei de confuzie să estimăm valorile TP , FP , TN , FN (și astfel $Precision/Recall$)? **să luăm exemplul pe c_2**

Măsuri de performanță (cont.)

> să considerăm cazul general, multiclassă, $c_i \in \{1, \dots, C\}$ (cont.);

Confusion Matrix (cont.)

c_2 : $TP = 6$, $FP = 14$, $FN = 1$

rezultat în urma clasificării

	c_1	c_2	c_i	c_C
c_1	5	2	3	1
c_2	0	6	1	0
c_i	1	2	11	3
c_C	1	10	0	3

> putem pe baza matricei de confuzie să estimăm valorile TP , FP , TN , FN (și astfel $Precision/Recall$)? **să luăm exemplul pe c_2**

Măsuri de performanță (cont.)

> să considerăm cazul general, multiclassă, $c_i \in \{1, \dots, C\}$ (cont.);

Confusion Matrix (cont.)

c_2 : $TP = 6$, $FP = 14$, $FN = 1$, $TN = 19$

rezultat în urma clasificării

	c_1	c_2	c_i	c_C
c_1	5	2	3	1
c_2	0	6	1	0
c_i	1	2	11	3
c_C	1	10	0	3

> putem pe baza matricei de confuzie să estimăm valorile TP , FP , TN , FN (și astfel $Precision/Recall$)? **să luăm exemplul pe c_2**

Măsurile de performanță (cont.)

> să considerăm cazul general, multclasă, $c_i \in \{1, \dots, C\}$ (cont.);

Confusion Matrix (cont.)

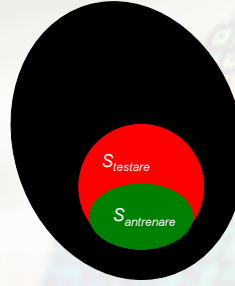
rezultat în urma clasificării

		c_1	c_2	c_i	c_C
realitate (ground truth)	c_1	5	2	3	1
	c_2	0	6	1	0
	c_i	1	2	11	3
	c_C	1	10	0	3

> cum determinăm Accuracy? =
 $(5 + 6 + 11 + 3) / (5 + 2 + 3 + 1 + 6 + 1 + 1 + 2 + 11 + 3 + 1 + 10 + 3)$

Evaluarea performanței

> clasificatorul este antrenat/optimizat pe date cunoscute;



date de clasificat
(dependente de aplicație)

= antrenăm pe $S_{antrenare}$, testăm performanța pe $S_{testare}$; modificăm clasificator/parametri până obținem cele mai bune rezultate;

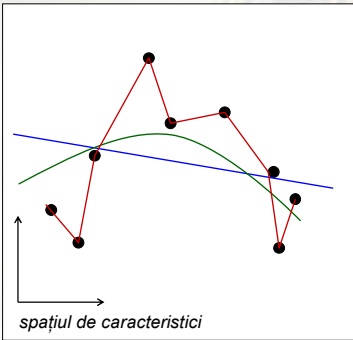
> cum alegem partiționarea setului cunoscut astfel încât să asigurăm **generalizarea maximă** pentru rezultatele obținute?

= clasificatorul se "va descurca" cu performanțe cel puțin superioare celor obținute pe datele cunoscute, pe datele reale, necunoscute.

Evaluarea performanței (cont.)

[Andrew W. Moore]

> să considerăm un exemplu particular: regresia;



[având la dispoziție un set de date, trebuie să determinăm ecuația care se potrivește cel mai bine acestora; "prezicem" astfel comportamentul datelor]

- regresie liniară;
- regresie pătratică;
- unim punctele;

care varianță este cea mai bună?

Evaluarea performanței (cont.)

[Andrew W. Moore]

% Split

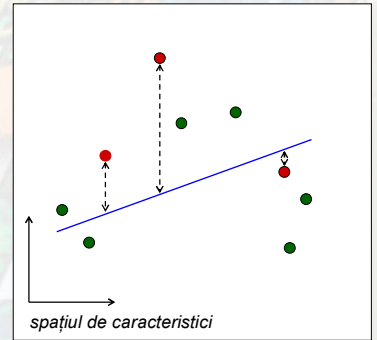
ex. 70% - $S_{antrenare}$ / 30% - $S_{testare}$

> datele sunt împărțite în mod aleator în $x\%$ pentru $S_{antrenare}$ și $(100-x)\%$ pentru $S_{testare}$;

> să reluăm exemplul anterior al regresiei;

> regresie liniară:

eroare pătratică medie = 2.4



Evaluarea performanței (cont.)

[Andrew W. Moore]

% Split (cont.)

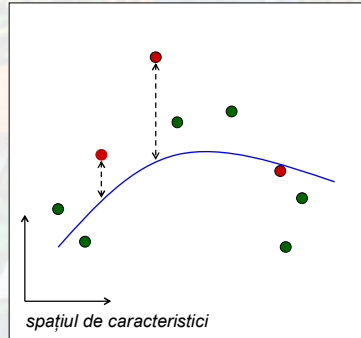
ex. 70% - $S_{antrenare}$ / 30% - $S_{testare}$

> datele sunt împărțite în mod aleator în $x\%$ pentru $S_{antrenare}$ și $(100-x)\%$ pentru $S_{testare}$;

> să reluăm exemplul anterior al regresiei (cont.);

> regresie pătratică:

eroare pătratică medie = 0.9



Evaluarea performanței (cont.)

[Andrew W. Moore]

% Split (cont.)

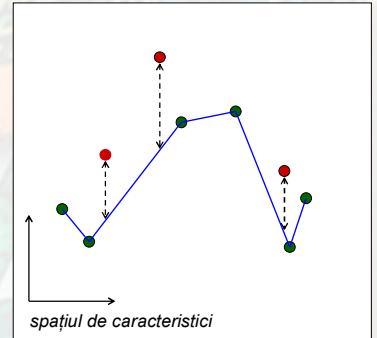
ex. 70% - $S_{antrenare}$ / 30% - $S_{testare}$

> datele sunt împărțite în mod aleator în $x\%$ pentru $S_{antrenare}$ și $(100-x)\%$ pentru $S_{testare}$;

> să reluăm exemplul anterior al regresiei (cont.);

> unire puncte:

eroare pătratică medie = 2.2

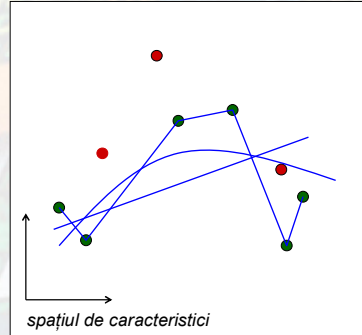


Evaluarea performanței (cont.)

% Split (cont.)

- > să reluăm exemplul anterior al regresiei (cont.);
- > care dintre variante oferă eroarea cea mai mică?
- > **“overfitting”** = clasificatorul învățat să se adapteze perfect datelor de antrenare; generalizare limitată.

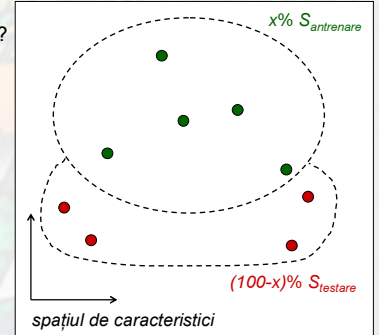
ex. 70% - $S_{antrenare}$ / 30% - $S_{testare}$



Evaluarea performanței (cont.)

% Split (cont.)

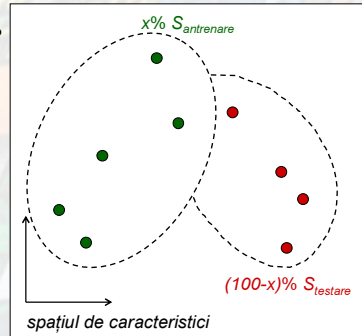
- > cum alegem $x\%$ date?
- > statistic, nu este suficient să testăm doar pentru o posibilă repartiție în $x\%$ (“do you feel lucky?”);
- > soluție: pentru $x\%$ fixat, alegem în mod aleator un anumit număr de repartiții $S_{antrenare} - S_{testare}$; mediere performanță.



Evaluarea performanței (cont.)

% Split (cont.)

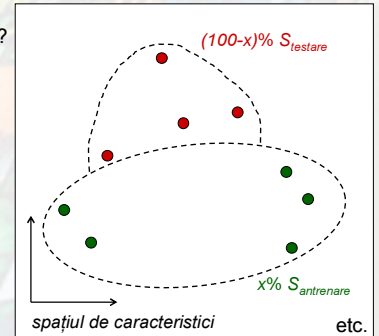
- > cum alegem $x\%$ date? (cont.)
- > soluție: pentru $x\%$ fixat, alegem în mod aleator un anumit număr de repartiții $S_{antrenare} - S_{testare}$; mediere performanță (cont.).



Evaluarea performanței (cont.)

% Split (cont.)

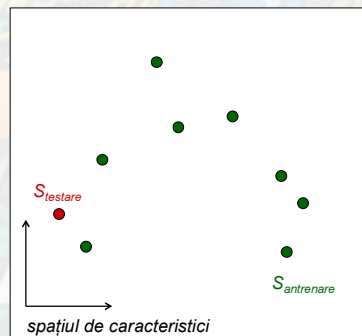
- > cum alegem $x\%$ date? (cont.)
- > soluție: pentru $x\%$ fixat, alegem în mod aleator un anumit număr de repartiții $S_{antrenare} - S_{testare}$; mediere performanță (cont.).



Evaluarea performanței (cont.)

Leave-one-out

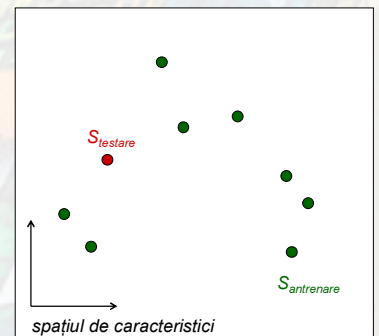
- > algoritm:
 - p1: parcurgem toate datele din setul cunoscut;
 - p2: eliminăm din set data curentă;
 - p3: antrenăm clasificator pe datele rămase;
 - p4: testăm pe data curentă (eliminată).



Evaluarea performanței (cont.)

Leave-one-out (cont.)

- > algoritm (cont.):
 - p1: parcurgem toate datele din setul cunoscut;
 - p2: eliminăm din set data curentă;
 - p3: antrenăm clasificator pe datele rămase;
 - p4: testăm pe data curentă (eliminată).

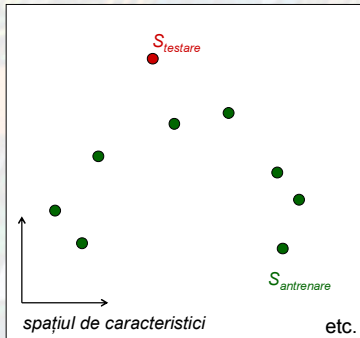


Evaluarea performanței (cont.)

Leave-one-out (cont.)

- > algoritm (cont.):
- p1: parcurgem toate datele din setul cunoscut;
- p2: eliminăm din set data curentă;
- p3: antrenăm clasificator pe datele rămase;
- p4: testăm pe data curentă (eliminată).

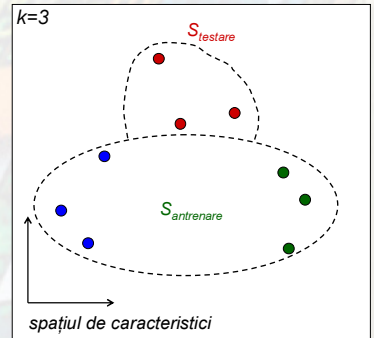
raportăm valoarea medie de performanță



Evaluarea performanței (cont.)

k-fold

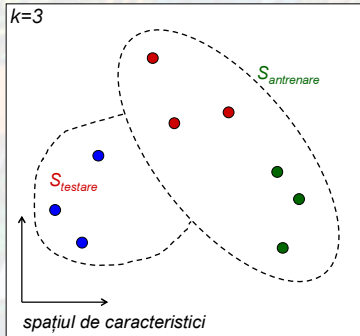
- > algoritm:
- p1: se alege k ;
- p2: datele se împart în k partiții (~egale);
- p3: se parcurg partițiile;
- p4: pentru partiția curentă, clasificatorul este antrenat pe celelalte date și testat pe această partiție.



Evaluarea performanței (cont.)

k-fold (cont.)

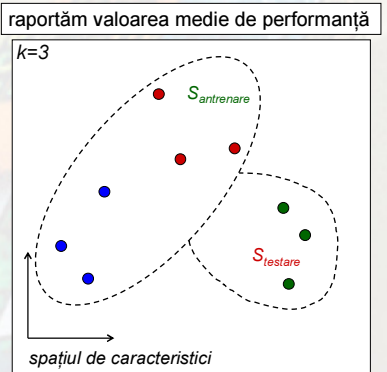
- > algoritm (cont.):
- p1: se alege k ;
- p2: datele se împart în k partiții (~egale);
- p3: se parcurg partițiile;
- p4: pentru partiția curentă, clasificatorul este antrenat pe celelalte date și testat pe această partiție.



Evaluarea performanței (cont.)

k-fold (cont.)

- > algoritm (cont.):
- p1: se alege k ;
- p2: datele se împart în k partiții (~egale);
- p3: se parcurg partițiile;
- p4: pentru partiția curentă, clasificatorul este antrenat pe celelalte date și testat pe această partiție.

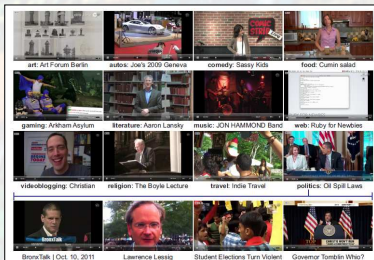


Exemple de sisteme de clasificare

determinare automată gen video

> *obiectiv*: realizarea unui sistem capabil să catalogheze automat genul video;

> experimentare date platformă *blip.tv*, 5.127 secvențe catalogate în 26 de genuri, ex. artă, auto, jurnalism, comedie, documentare, politică, religie, educație, sporturi, tehnologie, etc;



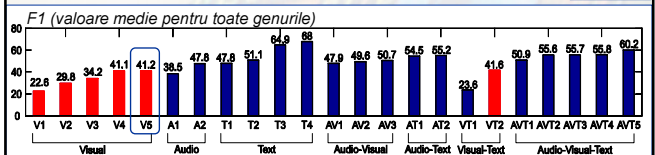
sursă blip.tv

[B. Ionescu et al., MediaEval 2012]

Exemple de sisteme de clasificare (cont.)

determinare automată gen video (cont.)

> % Split (antrenare 50% – testare 50%);



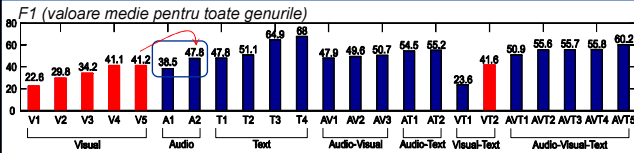
- capabilitate descriptori vizuali: 30%±10%;

- ce mai bună performanță este obținută pentru: LBP + Color Coherence Vector + histogram ($F1=41.2\%$).

Exemple de sisteme de clasificare (cont.)

determinare automată gen video (cont.)

> % Split (antrenare 50% – testare 50%);

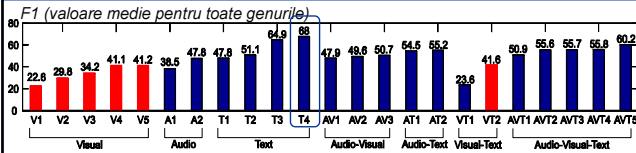


- folosirea informației audio se dovedește mai eficientă decât informația vizuală (creștere de ~6%);
- folosire descriptor audio bazat pe blocuri conduce la cele mai bune rezultate (cu ~10% > decât audio standard).

Exemple de sisteme de clasificare (cont.)

determinare automată gen video (cont.)

> % Split (antrenare 50% – testare 50%);

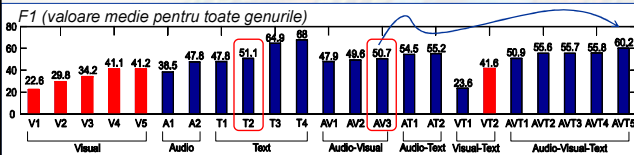


- la nivel de descriptori textuali, cea mai bună performanță pentru folosire ASR și metadatae blip.tv ($F1=68\%$).

Exemple de sisteme de clasificare (cont.)

determinare automată gen video (cont.)

> % Split (antrenare 50% – testare 50%);



- folosirea de descriptori audio-vizuali conduce la performanță apropiată de descriptori textuali (ASR);
- crescând numărul de modalități folosite conduce la creșterea semnificativă a performanței.

Exemple de sisteme de clasificare (cont.)

[J. Schlüter et al., MediaEval 2012]

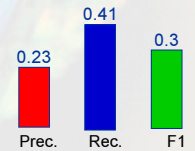
determinare automată conținut violență

- > *obiectiv*: realizarea unui sistem capabil să catalogheze automat conținutul video ca fiind violent sau nu;
- > experimentare producții tipice Hollywood, 15 filme;

Table 1: Evaluation of concept predictions

concept	vis.	aud.	dim.	prec.	rec.	F-sc.
blood	✓		5	0.07	1.00	0.12
coldarms	✓		1	0.11	1.00	0.19
firearms	✓		1	0.17	0.45	0.24
gore	✓		1	0.05	0.33	0.09
gunshots		✓	4	0.10	0.14	0.12
screams		✓	5	0.08	0.19	0.12
carchase	✓	✓	1	0.01	0.08	0.01
explosions	✓	✓	1	0.08	0.17	0.11
fighths	✓	✓	5	0.14	0.29	0.19
fire	✓	✓	1	0.24	0.30	0.26

> clasificator perceptron, evaluare leave-one-out;



Exemple de sisteme de clasificare (cont.)

recunoașterea automată a acțiunilor

- > *obiectiv*: realizarea unui sistem capabil să determine automat o serie de acțiuni umane din înregistrări video;

- > experimentare 6.600 filme YouTube ce conțin acțiuni uzuale precum mers cu bicicleta, cântat la chitară, exerciții fizice, parade, etc.



sursă YouTube

Exemple de sisteme de clasificare (cont.)

recunoașterea automată a acțiunilor (cont.)

- > evaluare 8-fold;

Table 2: Comparison with State-of-the-art on UCF50 Human Action Recognition.

Method	Accuracy
Reddy et al.	76.9%
SVM + FK vizual	74.7%
Solmaz et al.	73.7%
Everts et al.	72.9%
Kliper-Gross et al.	72.6%
Solmaz et al.	65.3%

[I. Mironică et al., ACM MM 2013]



> Sfârșit M5