LAPI – Laboratorul de Analiza şi Prelucrarea Imaginilor

Universitatea POLITEHNICA din Bucureşti

Facultatea de Electronică, Telecomunicaţii şi Tehnologia Informaţiei

TACAI - Tehnici de Analiză și Clasificare Automată a Informației

Note de laborator

Dr.ing. Ionuț Mironică

Conf.dr.ing. Bogdan Ionescu

Laborator 1

Cuprins

- Introducere în algoritmi de clasificare
- Introducere Weka
- Evaluarea performanței de clasificare
- Exerciții

I. Introducere

Ce este Machine learning?

- **Exemplu aplicație:** Este foarte greu de scris un algoritm care recunoaşte un set de gesturi statice / dinamice sau care să rezolve problema de recunoaştere a feţei:
  - Nu ştim cum funcţionează creierul uman pentru a clasifica gesturile;
  - Chiar dacă am şti nu am avea idee cum să programăm deoarece ar fi foarte complicat;
  - Ar trebui să scriem o funcție diferită pentru fiecare gest.

- În loc să scriem programe foarte multe, putem colecta exemple care specifică fiecare gest;

- Un algoritm de învăţare va prelua aceste exemple şi va “creea” un program care va face această clasificare în mod automat;

I. Introducere

Ce este Machine learning?

- Există mii de algoritmi de învăţare / sute dintre ei apar anual;
- Există mai multe tipuri de învăţare:

Învățare supervizată

- datele de antrenare conţin şi ieşirea dorită;

Învățare nesupervizată

- datele de antrenare nu conţin ieşirea dorită (clusterizare);
- ideea de bază este de a se găsi şabloane şi pattern-uri în date care să fie evidenţiate în mod automat.

Învăţare semi-supervizată

- doar o parte din datele de antrenare conţin ieşirea dorită;

Reinforcement Learning

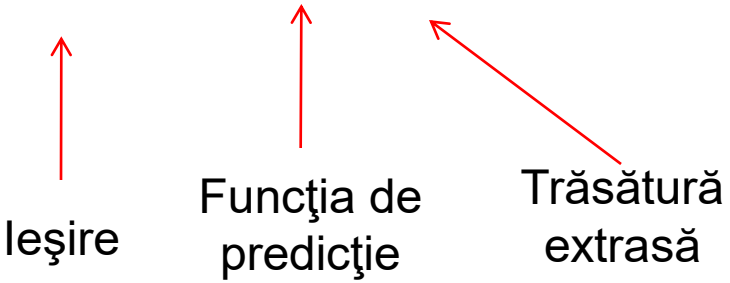
- se învaţă în funcţie de feedback-ul primit după ce o decizie este luată.

I. Introducere

Învăţare supervizată

Model de bază

$$y = f(\mathbf{x})$$



Antrenare: fiind dată o mulţime de antrenare împreună cu răspunsul dorit $\{(\mathbf{x}_1,y_1), \ldots, (\mathbf{x}_N,y_N)\}$, se estimează predicţia funcţiei f prin minimizarea erorii de predicţie pe mulţimea de antrenare;

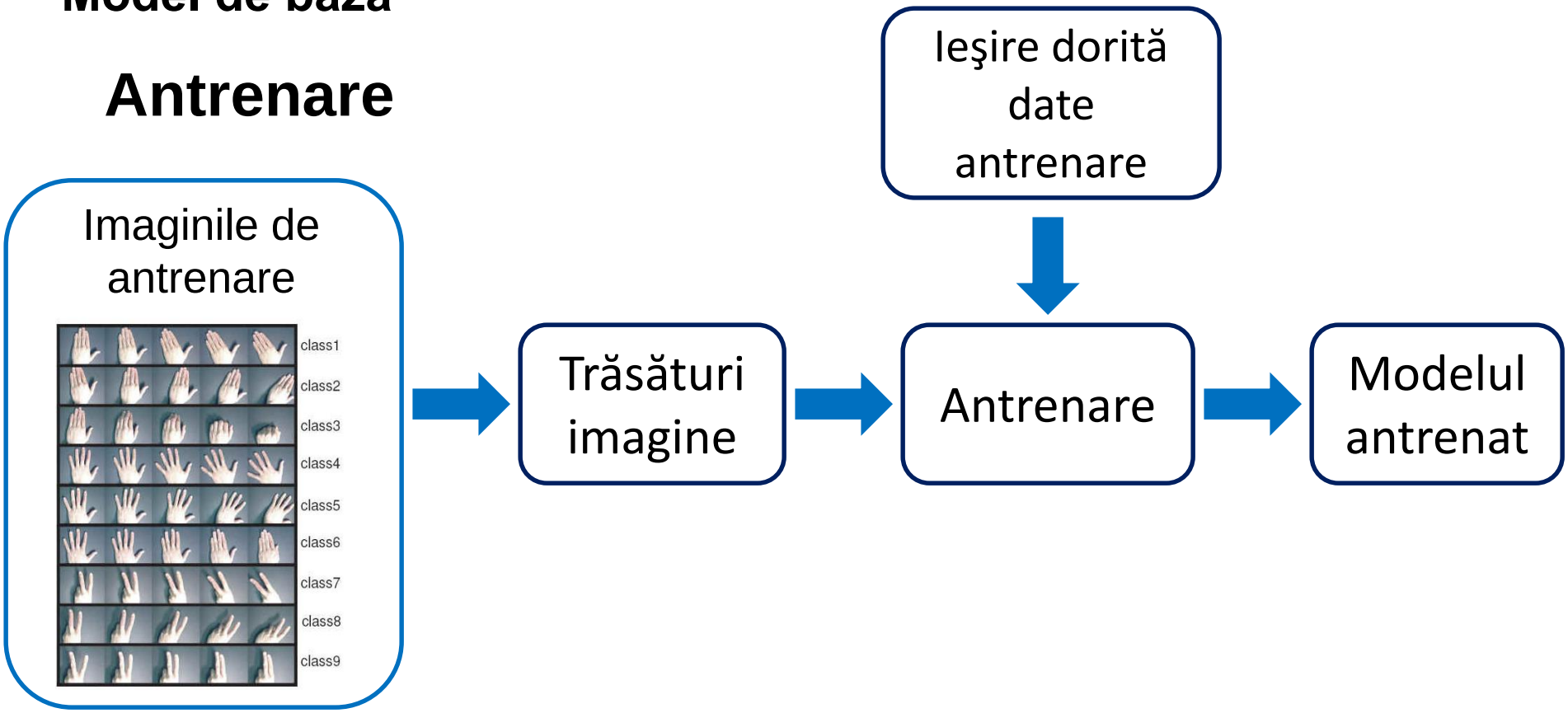
Testare: se aplică funcţia f pe un exemplu de test $\mathbf{x}$(care nu a fost folosit în procesul de antrenare) şi prezintă ieşirea funcţiei $y = f(\mathbf{x})$.

I. Introducere

Învăţare supervizată

Model de bază

Antrenare



I. Introducere

Învăţare supervizată

Model de bază

Testare

Extragere trăsături (X)

Aplicare funcţie de predicţie f(x)

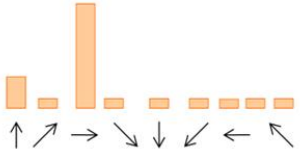
Predicţie

I. Introducere

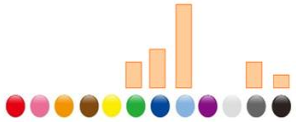
Învăţare supervizată

Ce sunt trăsăturile?

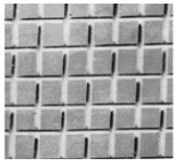
pixeli

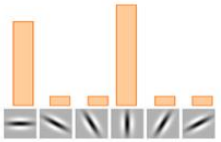


muchii



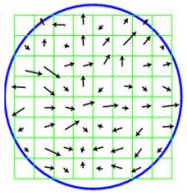
culoare





textură

formă

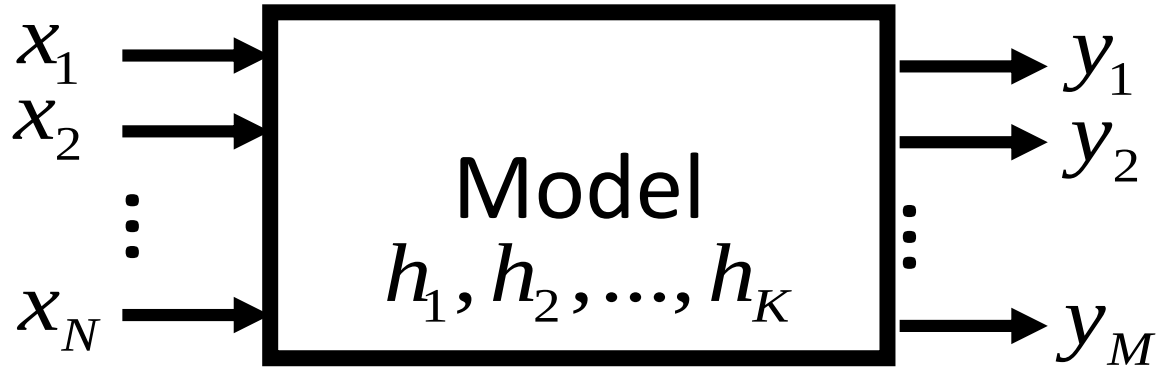


puncte de interes

I. Introducere

Învăţare supervizată

Învățare supervizată – schemă de bază



Variabile de intrare: $\mathbf{x} = (x_1, x_2, \ldots, x_N)$

Variabile ascunse: $\mathbf{h} = (h_1, h_2, \ldots, h_K)$

Variabile de ieşire: $\mathbf{y} = (y_1, y_2, \ldots, y_K)$

I. Introducere

Învăţare supervizată

Algoritmi existenţi

- Support vector machines (SVM),
- Deep learning,
- Rețele recurente,
- Reţele bayesiene,
- Random forest,
- K-nearest neighbor (k-NN),

Etc.

Care este cel mai bun algoritm?

I. Introducere

Învăţare supervizată

Teorema “No free lunch”

[https://cynicalbabblings.wordpress.com]

II. Weka

Informații generale

- Reprezintă o colecție de algoritmi de Machine Learning pentru diferite probleme de data-mining;
- Dezvoltat de către Universitea din Waikato, Noua Zeelandă;
- Open-Source dezvoltat in JAVA (GNU General Public License);
- Trăsături principale:
  - intrumente de preprocesare,
  - algoritmi de învățare,
  - algoritmi de clusterizare,
  - reguli de asociere,
  - metode de evaluare,
  - interfață grafică,
  - instrumente pentru comparația clasificatorilor.

II. Weka

Documentație

- **Site-ul Weka:**

http://www.cs.waikato.ac.nz/~ml/weka/

- **Documentație Weka:**

http://transact.dl.sourceforge.net/sourceforge/weka/WekaManual-3.6.0.pdf

http://www.cs.waikato.ac.nz/ml/weka/documentation.html

- **Tutoriale video:**

https://weka.waikato.ac.nz/explorer

II. Weka

Interfață

Exporer:

- reprezintă componenta principală vizuală din Weka. Conține mai multe componente:
  - *Preprocess*: opțiuni de încărcare a bazelor de date (format arff / csv) / conectare Sql / procesări atașate datelor.
  - *Clasiffy*: algoritmi de clasificare;
  - Associate: algoritmi de asociere;
  - *Cluster*: algoritmi de învățare nesupervizată
  - *Select attributes*: algoritmi de detecție a atributelor
  - *Visualize:* modalități de vizualizare a datelor.

Experimenter:

- permite configurarea unui sistem ce asigură comparația sistematică a perfomanțelor algorimilor de clasificare pe diferite colecții de baze de date.

KnowledgeFlow:

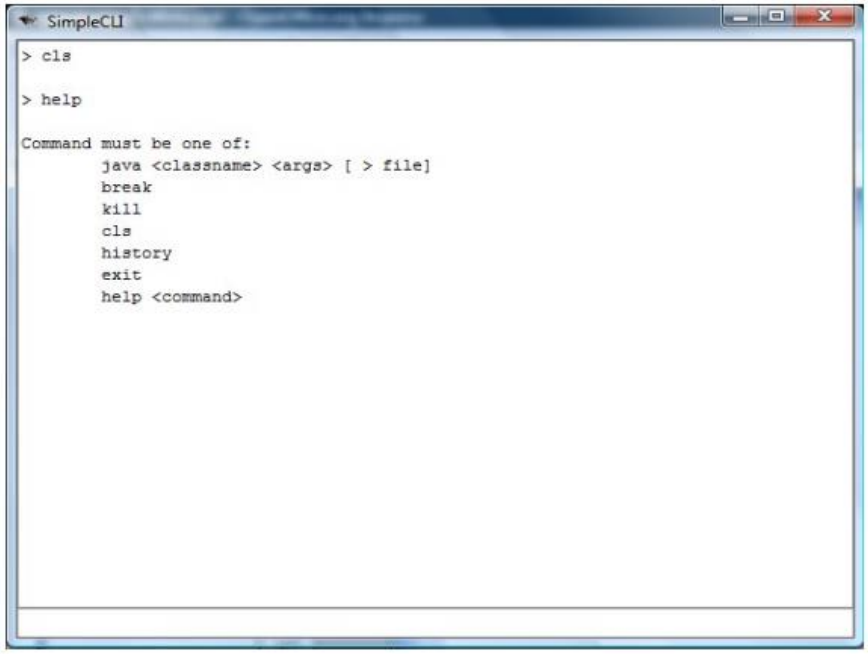
- permite configurarea de fluxuri automate.

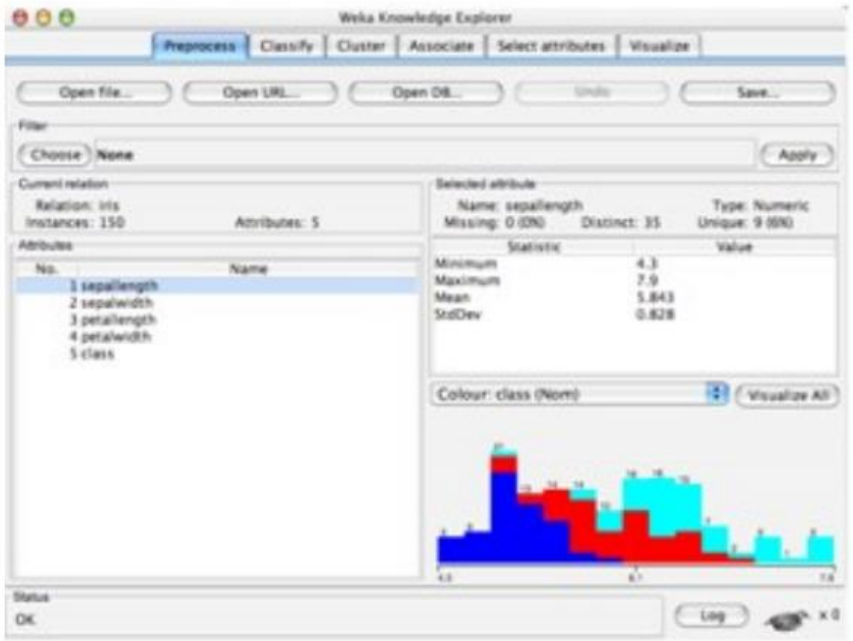
SimpleCLI:

- consolă de apelare a funcțiilor de clasificare.

II. Weka

CLI vs GUI





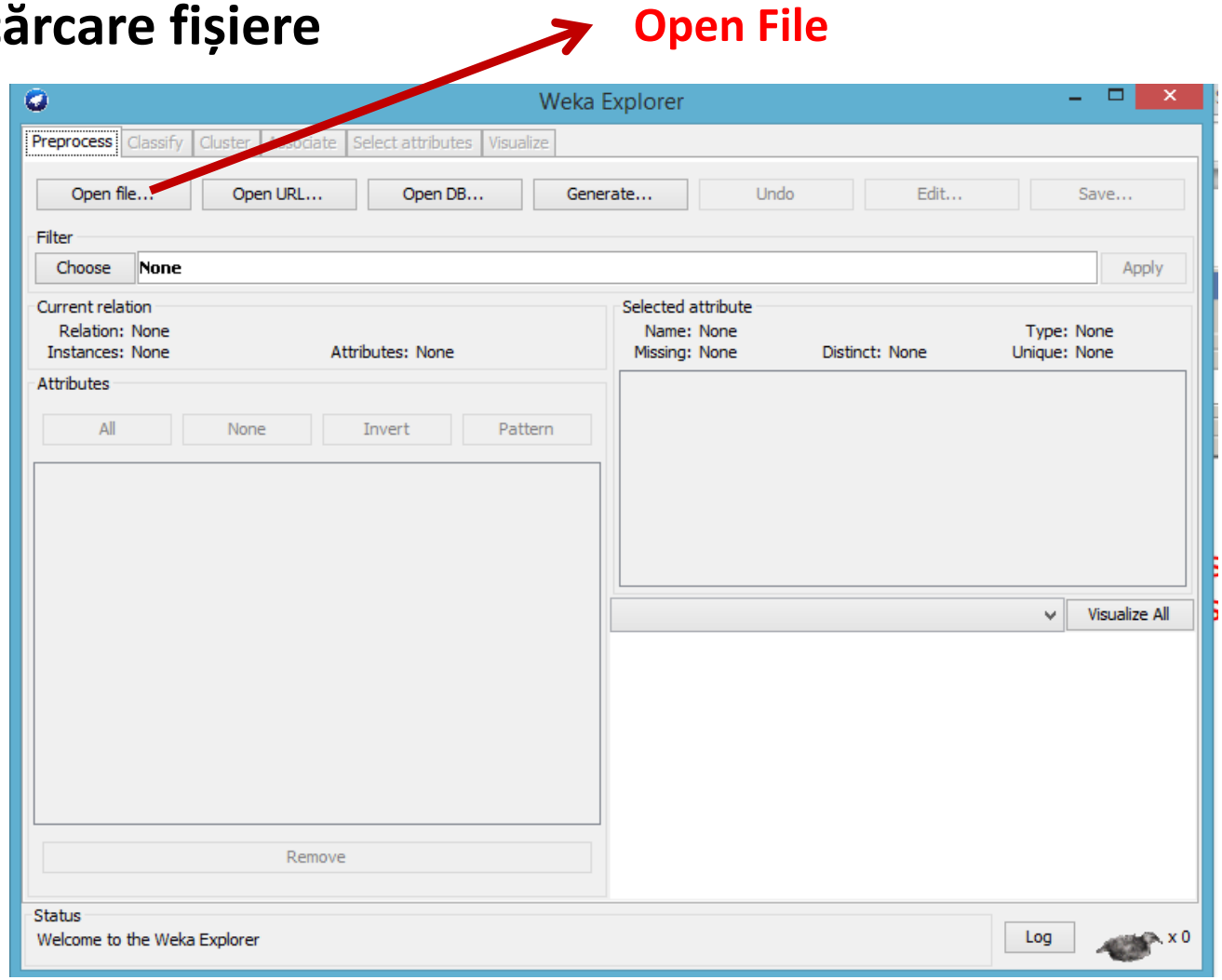
- Recomandat pentru utilizarea în profunzime a algoritmilor;
- Oferă funcționalități indisponibile în interfața grafică.

- Mai ușor de utilizat;
- Oferă funcționalități intuitive: Explorer, Experimenter și KnowledgeFlow.

II. Weka

Explorer: Preprocess

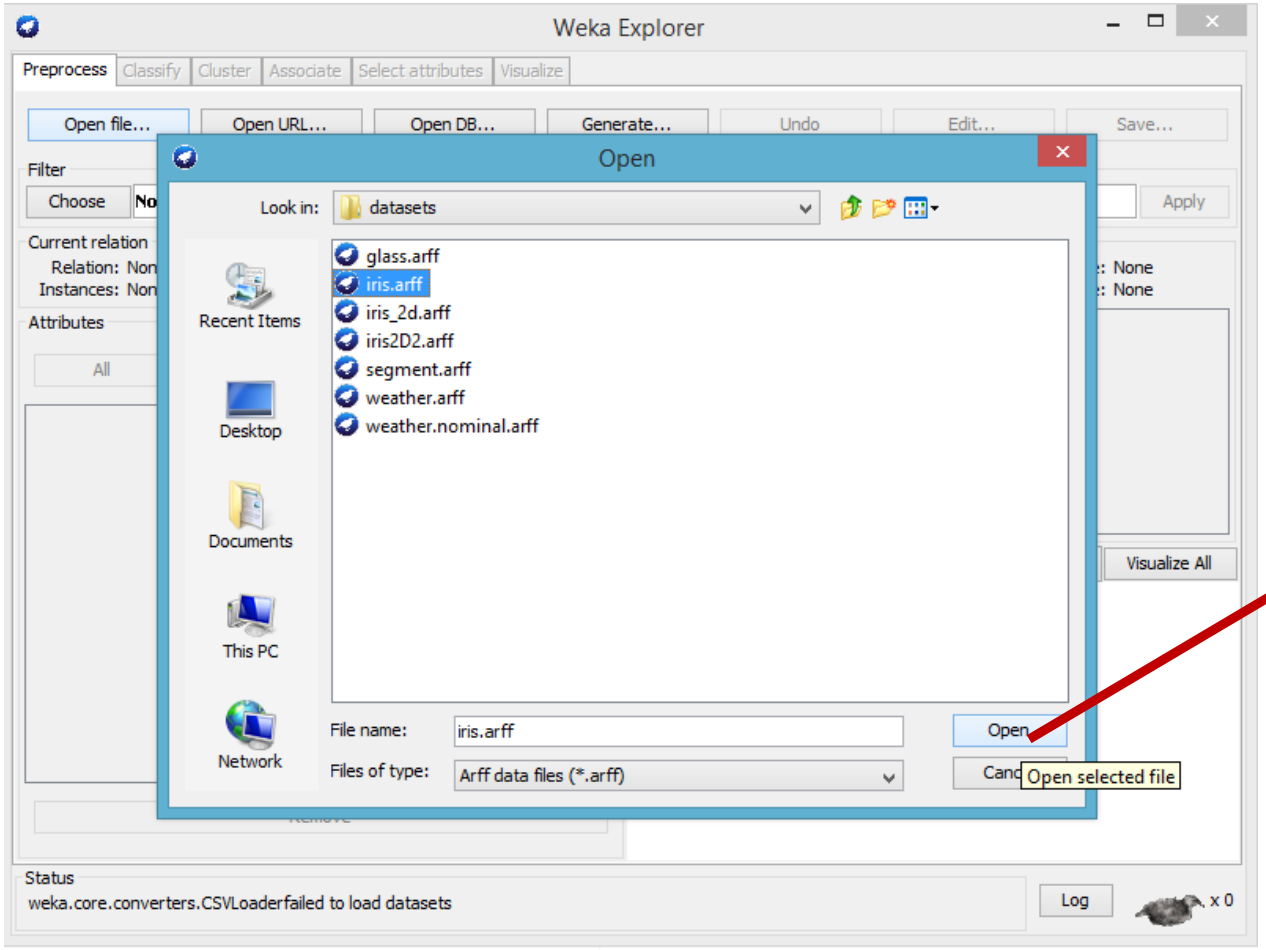
Încărcare fișiere



II. Weka

Explorer: Preprocess

Încărcare fișiere



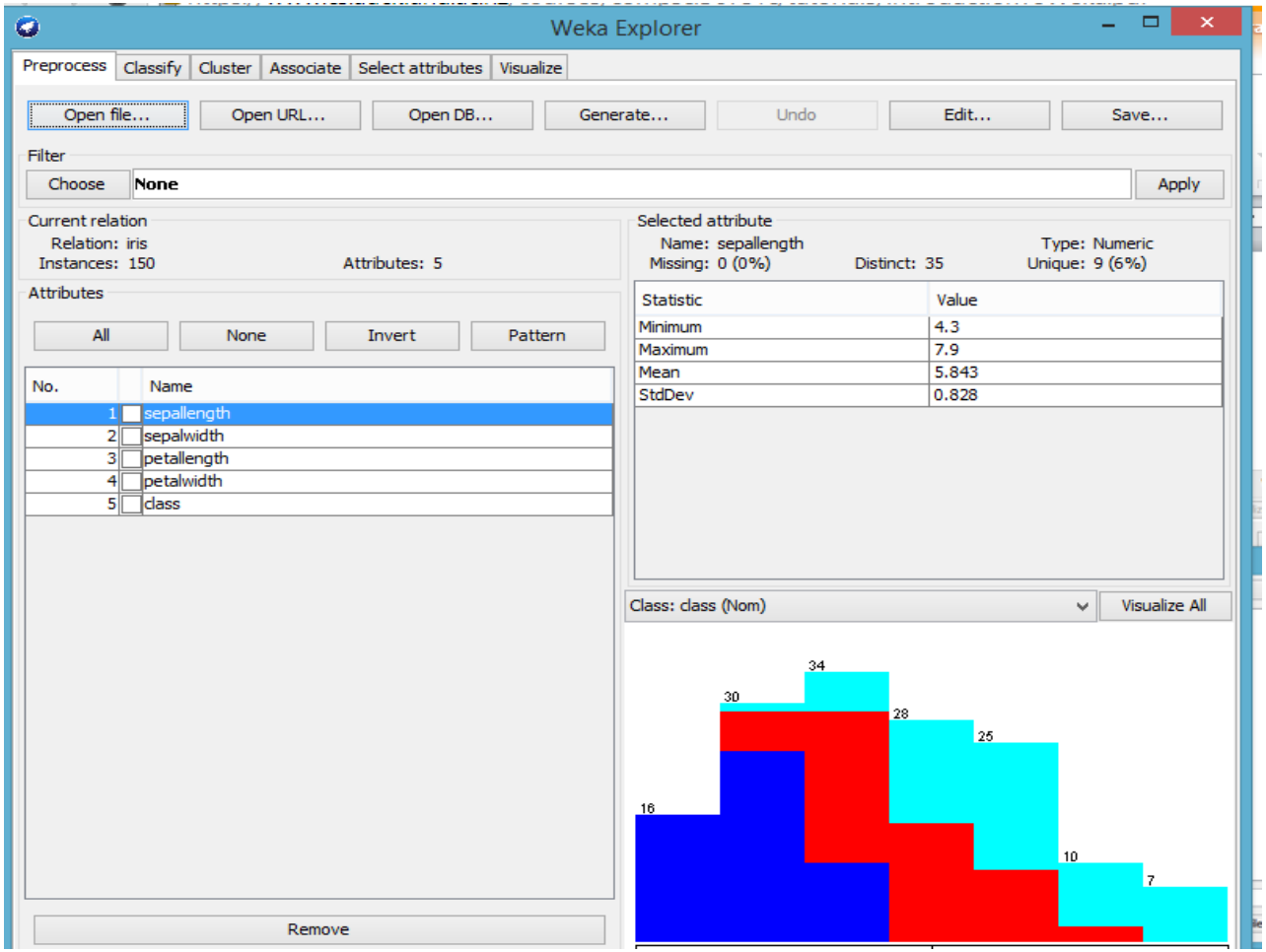
Open:

Selecție fișier (arff / csv)

II. Weka

Explorer: Preprocess

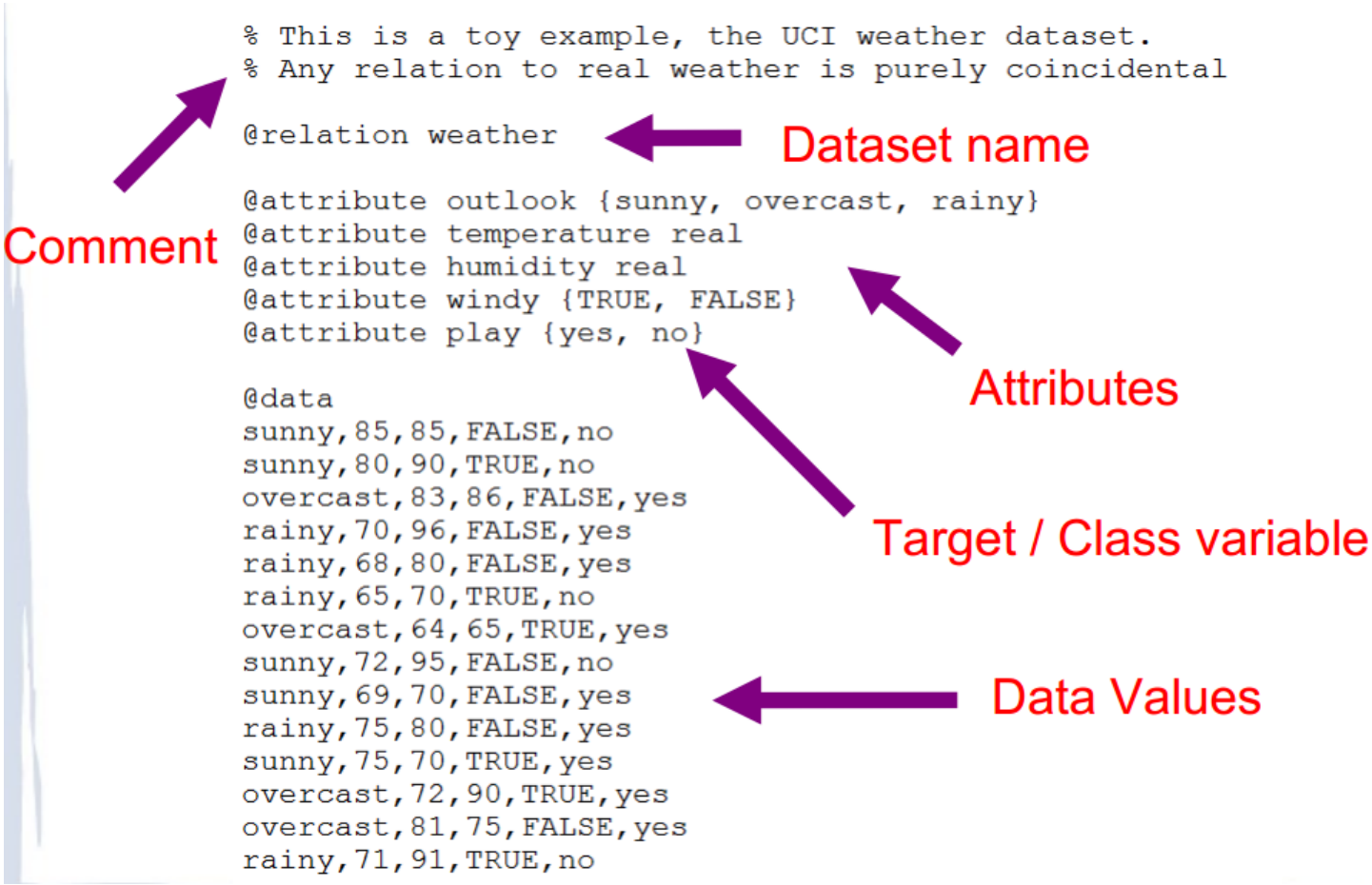
Încărcare fișiere



II. Weka

Explorer: Preprocess

Structură fișier arff



II. Weka

Explorer: Preprocess

Atribute fișier arff

- **Atribute nominale**: valorile sunt selectate dintr-o listă predefinită;
- **Atribute numerice:** valorile sunt întregi sau reale;
- **Șiruri de caractere (string):** sunt încadrate între ghilimele;
- **Relaționale:** pentru date care aparțin în mai multe tipuri de instanțe.

II. Weka

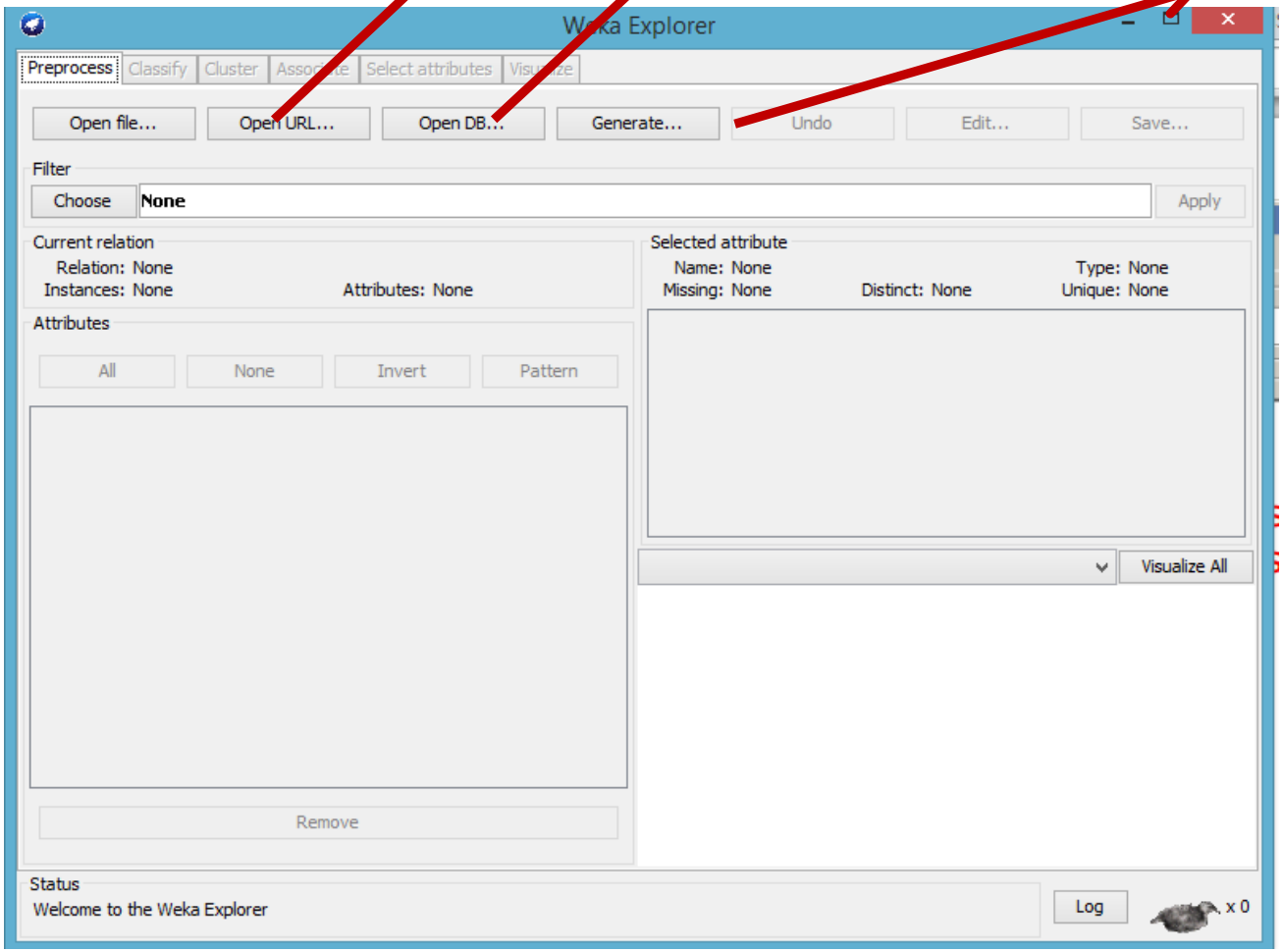
Explorer: Preprocess

Încărcare fișiere

Open URL: încărcare fișiere aflate la un link;

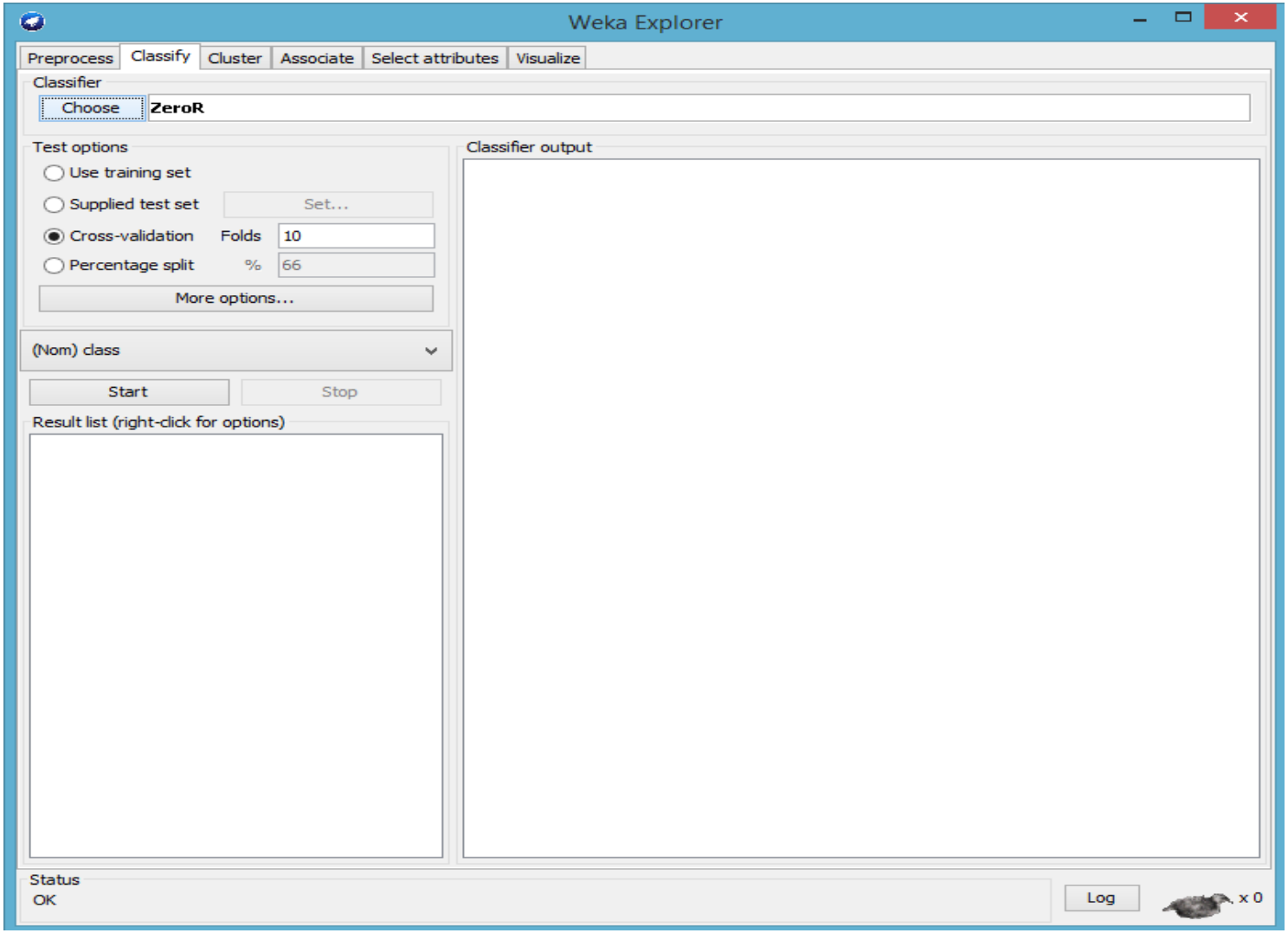
Open DB: conectare la o baza de date relațională (SQL) prin driver JBDC;

Generate: generare baze de date.



II. Weka

Utilizare algoritmi de clasificare



II. Weka

Explorer: Preprocess

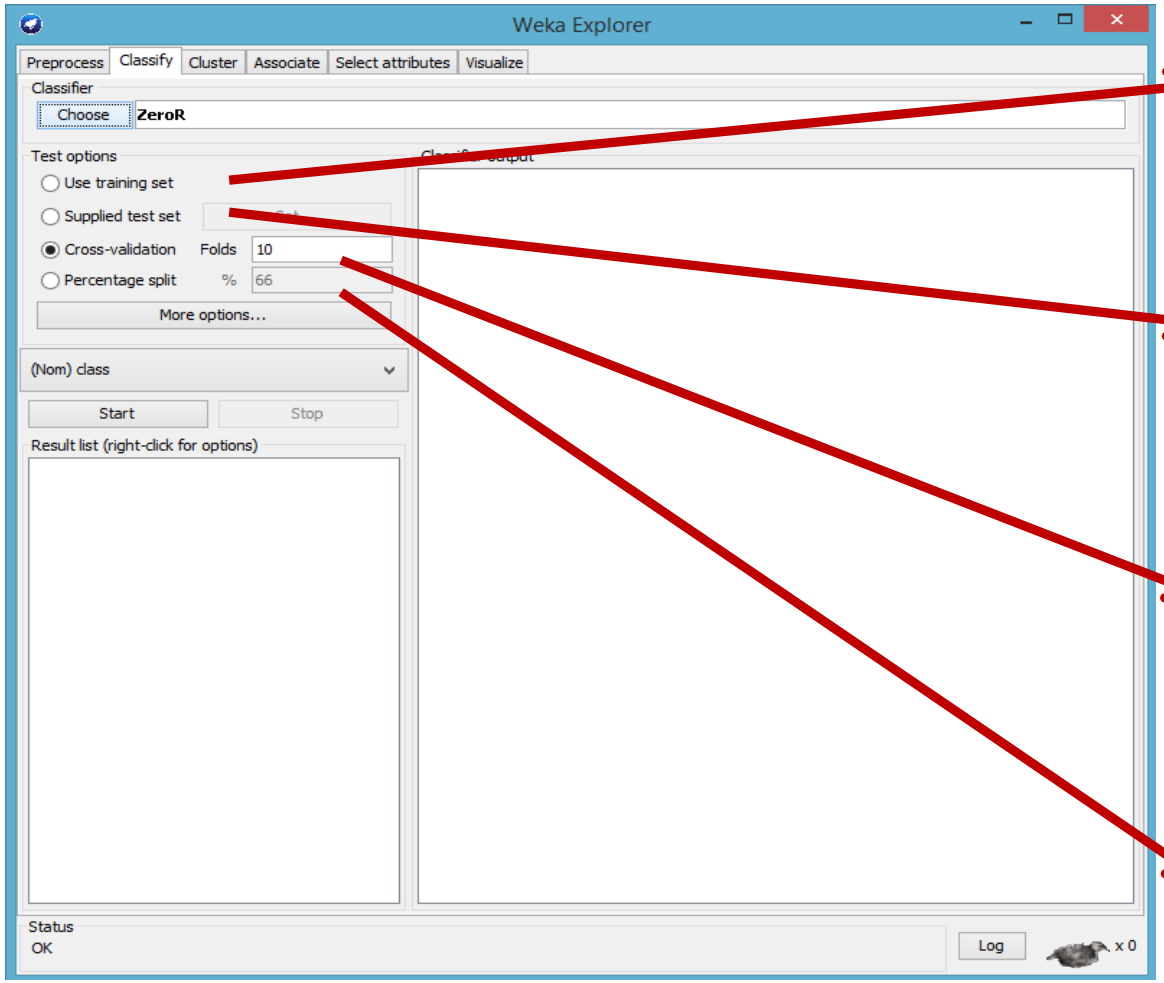
Încărcare fișiere

1. Deschide weka GUI
2. Click 'Explorer'
3. 'Open file...'
4. Selectează tabul 'Classify' Alege un clasificator
5. Selectează parametrii de clasificare
7. Click 'Start'
8. Wait...
9. Vizualizează rezultate
    a. ‘Salvează rezultat'
    b. 'Salvează clasificator'

II. Weka

Utilizare algoritmi de clasificare

Opțiuni de împărțire a bazei de date (Test Options)



Use training set: utilizarea setului de antrenare și pentru testare;

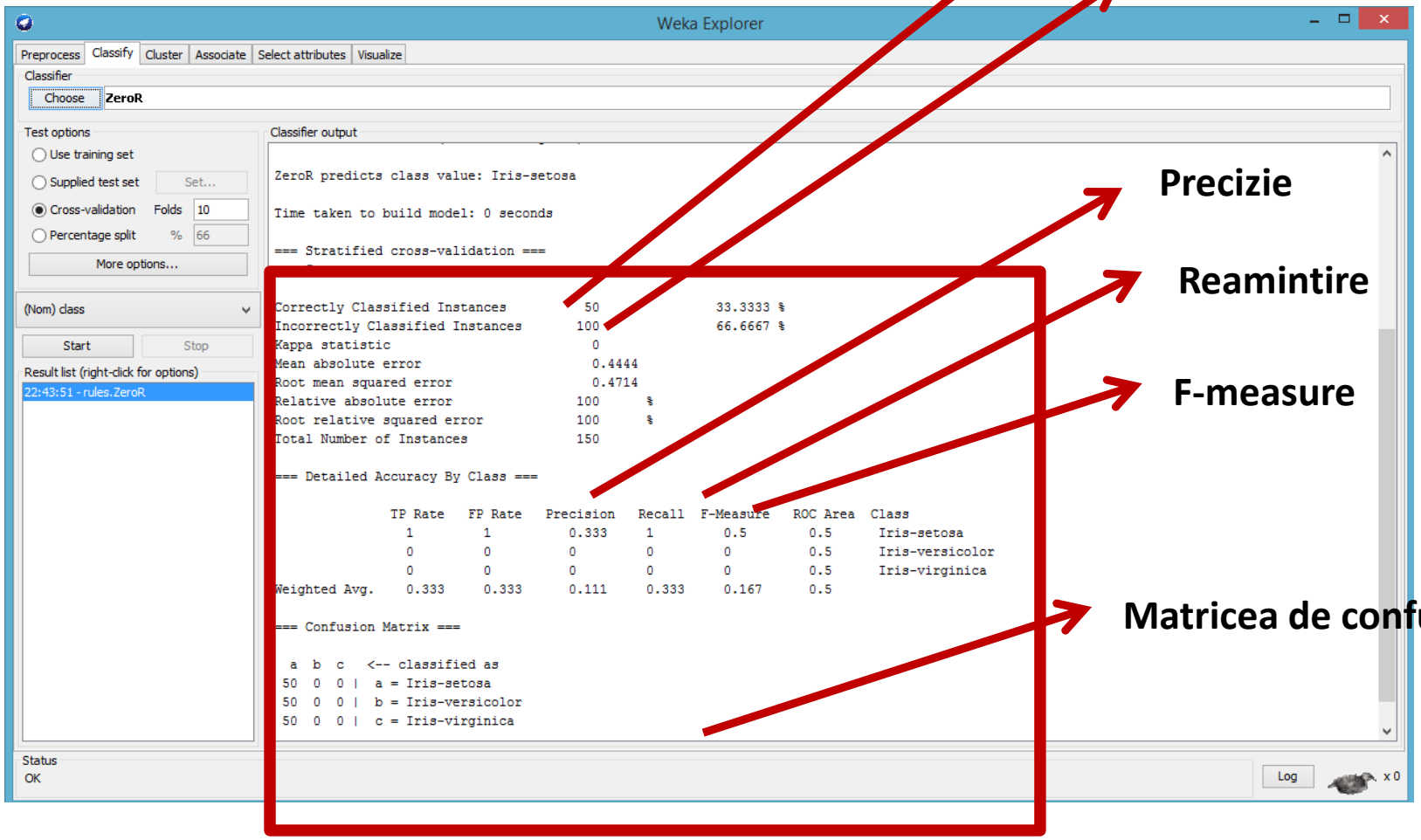
Supplied testset: baza de date de test încărcată separat;

Cross-validation: împărțirea setului de antrenare în mai multe seturi succesive de antrenare / testare;

Percentage split: împărțirea setului de antrenare într-un set de de antrenare și testare.

III. Evaluarea performanței de clasificare

Parametrii de evaluare



III. Evaluarea performanței de clasificare

Acuratețea de clasificare

Acuratețea se calculeaza ca raportul dintre numarul de date etichetate corect împărțit la numărul total de exemple.

$$Acuratețea = \frac{TP+TN}{TP+TN+FN+FP}$$

$$Procentul\ de\ clasificări\ incorecte = \frac{FN+FP}{TP+TN+FN+FP}$$

$$= 1 - \text{acuratețe}$$

	Relevant	Nerelevant
Regăsit	True positive (TP)	False pozitive (FP)
Neregăsit	False negative (FN)	True negative (TN)

III. Evaluarea performanței de clasificare

Acuratețea de clasificare

Dezavantaje in cazurile:

- Clase neechilibrate ca numar de date pentru fiecare (clase nebalansate);

- Costuri diferite de clasificare greșită pentru fiecare clasa în parte.

III. Evaluarea performanței de clasificare

Precizia și reamintirea

- **Precizia (precision)**: procentul de documente returnate care sunt și relevante.
- **Reamintirea (recall)**: procentul de documente relevante care sunt și regăsite.

	Relevant	Nerelevant
Regăsit	True positive (TP)	False pozitive (FP)
Neregăsit	False negative (FN)	True negative (TN)

- Precizia = tp/(tp + fp)
- Reamintirea = tp/(tp + fn)

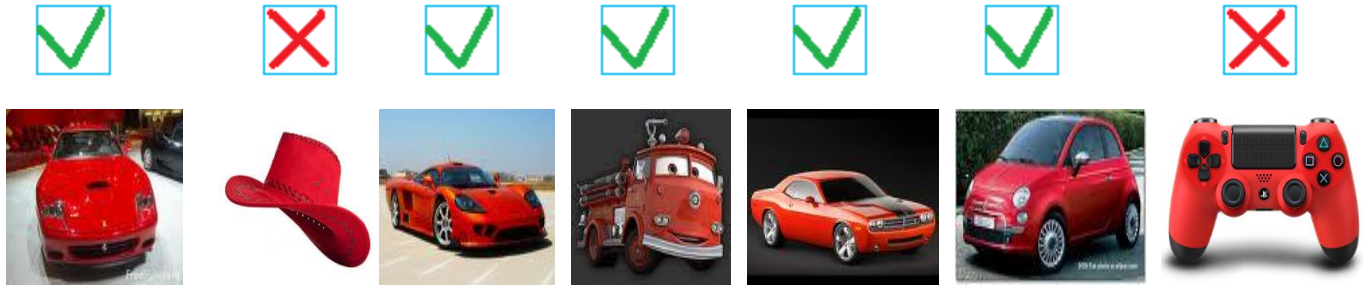
III. Evaluarea performanței de clasificare

Precizia și reamintirea

- Precizia = tp/(tp + fp)
- Reamintirea = tp/(tp + fn)

	Relevant	Nerelevant
Regăsit	True positive (TP)	False pozitive (FP)
Neregăsit	False negative (FN)	True negative (TN)

Relevanță



Rezultate

- Calculați precizia și reamintirea pentru această situație.

III. Evaluarea performanței de clasificare

Precizia și reamintirea

- **Întrebări**:

1) Ce este mai important? Să avem o precizie ridicată sau o reamintire ridicată?

2) Se poate să avem reamintire mare și precizie foarte mică?

III. Evaluarea performanței de clasificare

F-Measure

- **F-measure combină precizia cu reamintirea (medie armonică ponderată)**:

$$F = \frac{1}{\alpha \frac{1}{P} + (1-\alpha)\frac{1}{R}} = \frac{(\beta^2+1)PR}{\beta^2 P + R}$$

- De obicei se selectează F_1 measure
  - $\beta = 1$ sau $\alpha = ½$
  - Putem alege și alte variante?

III. Evaluarea performanței de clasificare

Matricea de confuzie

- Pentru *k* clase, este o matrice de *k* × *k*
- Elementul de la pozitia *(i, j)* arată numărul de date care fac parte din clasa *i* iar clasificatorul le etichetează cu clasa *j*.
- Tabelul următor ilustrează cazul particular pentru trei clase

		Clasa prezisă		
		Clasa 1	Clasa 2	Clasa 3
Clasa adevărată	Clasa 1	3	0	0
	Clasa 2	0	2	1
	Clasa 3	1	0	3

III. Evaluarea performanței de clasificare

Matricea de confuzie

		Clasa prezisă		
		Clasa 1	Clasa 2	Clasa 3
Clasa adevărată	Clasa 1	3	0	0
	Clasa 2	0	2	1
	Clasa 3	1	0	3

$$Acuratețea = \frac{8}{10} = 80\%$$

IV. Preprocesarea trăsăturilor

Studiu de caz

Detecția persoanelor care nu își plătesc creditul

Starea contului curent	**Istoric credit**	**Țară**	**Vârstă**	**Salariu**	**Telefon**
0 <= ... < 200 EUR	Fără istoric credite	Germania	22	1000	nu
>= 2000 EUR	Toate creditele plătite integral	Franța	38	500	nu
0 <= ... < 200 EUR	Întârzieri de plată		28		da
200 <= ... < 2000 EUR		Tailanda	55	100000	da
>= 2000 EUR	Fără istoric credite		55	1000	

IV. Preprocesarea trăsăturilor

Prelucrare date lipsă

Starea contului curent	Istoric credit	Țară	Age	Salariu	Telefon
0 <= ... < 200 EUR	Fără istoric credite	Germania	22	1000	nu
>= 2000 EUR	Toate creditele plătite integral	Franța	38	500	nu
0 <= ... < 200 EUR	Întârzieri de plată	?	28	?	da
200 <= ... < 2000 EUR	?	Tailanda	55	100000	da
>= 2000 EUR	Fără istoric credite	?	55	1000	?

IV. Preprocesarea trăsăturilor

Prelucrare date lipsă

Varianta 1: eliminare linii

Alternativa 2:

Utilizarea unor date prelucrate:

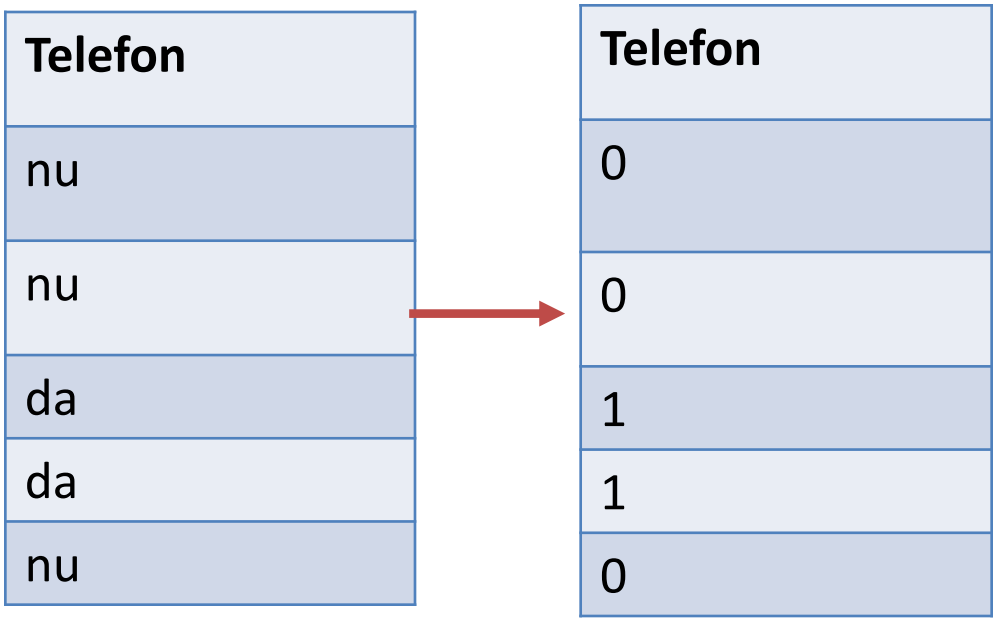
- Medie, valoare mediană, valoare minimă și maximă
- Cea mai utilizată valoare
- Adăugarea unei valori noi pentru date lipsă

Alternativa 3: Realizarea unui model care să estimeze valoarea lipsă

IV. Preprocesarea trăsăturilor

Valori nominale

Varianta 1: binarizarea valorilor



Telefon
nu
nu
da
da
nu

Telefon
0
0
1
1
0

IV. Preprocesarea trăsăturilor

Valori nominale

Varianta 2

Label Encoder: aplică valori între **0** și **nr_clase-1**.

Starea contului curent	
0 <= ... < 200 EUR	1
>= 2000 EUR	2
0 <= ... < 200 EUR	
200 EUR <= ... < 2000 EUR	3
>= 2000 EUR	

Starea contului curent	
0 <= ... < 200 EUR	1
>= 2000 EUR	3
0 <= ... < 200 EUR	
200 EUR <= ... < 2000 EUR	2
>= 2000 EUR	

IV. Preprocesarea trăsăturilor

Valori nominale

Varianta 3

Dummy Encoder: realizează **n** coloane pentru fiecare categorie.

În Weka această procesare se poate aplica utilizând opțiunea:

Filter -> filters -> unsuppervised -> attribute -> NominalToBinary

Istoric credit
Fără istoric credite
Toate creditele plătite integral
Întârzieri de plată
Toate creditele plătite integral
Fără istoric credite

Fără istoric credite	Toate creditele plătite integral	Întârzieri de plată
1	0	0
0	1	0
0	0	1
0	1	0
1	0	0

IV. Preprocesarea trăsăturilor

Valori nominale

Dummy Encoder: În cazul în care sunt prea multe coloane, aceasta variantă nu este fezabilă deoarece numărul de trăsături va fi foarte ridicat

Țară
Germania
Franța
Africa de Sud
Tailanda
SUA

Germania	Franța
1	0
0	1
0	0
0	0
0	0

250 coloane

SUA
0
0
0
0
1

IV. Preprocesarea trăsăturilor

Valori nominale

Posibilă soluție: clusterizarea categoriilor

Țară
Germania
Franța
Africa de Sud
Tailanda
SUA

→

Regiune
Europa
Europa
America
Asia
America

→

Europa	Africa	Asia	America
1	0	0	0
1	0	0	0
0	1	0	0
0	0	1	0
0	0	0	1

IV. Preprocesarea trăsăturilor

Game de valori diferite

Vârstă	Salariu
22	1000
38	500
28	1000000
55	100000
55	1000

[18 - 80] [0 - 1000000]

Vârstă	Salariu
0.22	0.001
0.38	0.0005
0.28	1
0.55	0.1
0.55	0.001

Normalizare

$$x' = \frac{x - \text{mean}(x)}{\max(x) - \min(x)}$$

În Weka această procesare se poate aplica utilizând opțiunea:
Filter -> filters -> unsuppervised -> attribute -> Normalize

V. Modele de Machine Learning

ZeroR

Cel mai simplu clasificator din Weka:

- Determină care este cea mai des întâlnită clasă
- În caz de regresie, va utiliza valoarea mediană
- Poate fi utilizată ca și limită inferioară a performanței pe care o poate primi un algoritm

V. Modele de Machine Learning

ZeroR

Predictors				Target
Outlook	Temp.	Humidity	Windy	Play Golf
Rainy	Hot	High	False	No
Rainy	Hot	High	True	No
Overcast	Hot	High	False	Yes
Sunny	Mild	High	False	Yes
Sunny	Cool	Normal	False	Yes
Sunny	Cool	Normal	True	No
Overcast	Cool	Normal	True	Yes
Rainy	Mild	High	False	No
Rainy	Cool	Normal	False	Yes
Sunny	Mild	Normal	False	Yes
Rainy	Mild	Normal	True	Yes
Overcast	Mild	High	True	Yes
Overcast	Hot	Normal	False	Yes
Sunny	Mild	High	True	No

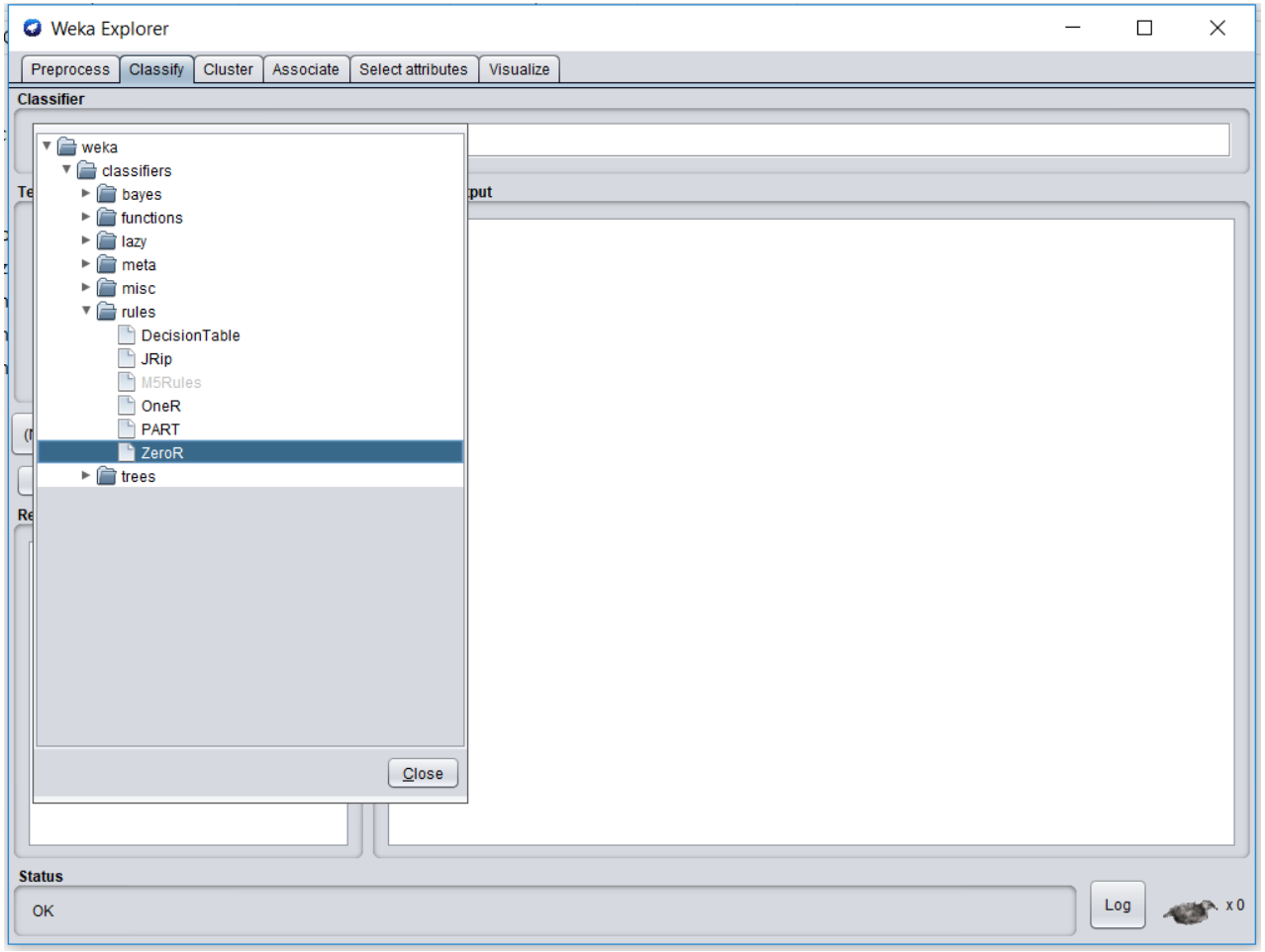
Play Golf	
Yes	No
9	5

Toate instanțele vor fi clasificate în clasa „Yes”

V. Modele de Machine Learning

ZeroR

În imagine se poate vizualiza opțiunea de aplicare a algoritmului ZeroR în Weka.



V. Modele de Machine Learning

OneR

OneR („One Rule”) reprezintă o metodă naivă de clasificare care generează o singură regulă utilizând o singură trăsătură.

Algoritmul găsește trăsătura care maximizează probabilitatea de clasificare.

V. Modele de Machine Learning

OneR

Which one is the best predictor ?

Outlook	Temp	Humidity	Windy	Play Golf
Rainy	Hot	High	False	No
Rainy	Hot	High	True	No
Overcast	Hot	High	False	Yes
Sunny	Mild	High	False	Yes
Sunny	Cool	Normal	False	Yes
Sunny	Cool	Normal	True	No
Overcast	Cool	Normal	True	Yes
Rainy	Mild	High	False	No
Rainy	Cool	Normal	False	Yes
Sunny	Mild	Normal	False	Yes
Rainy	Mild	Normal	True	Yes
Overcast	Mild	High	True	Yes
Overcast	Hot	Normal	False	Yes
Sunny	Mild	High	True	No

Frequency Tables

★		Play Golf	
		Yes	No
Outlook	Sunny	3	2
	Overcast	4	0
	Rainy	2	3

		Play Golf	
		Yes	No
Temp.	Hot	2	2
	Mild	4	2
	Cool	3	1

		Play Golf	
		Yes	No
Humidity	High	3	4
	Normal	6	1

		Play Golf	
		Yes	No
Windy	False	6	2
	True	3	3

În final vom avea următoarea regulă generată:

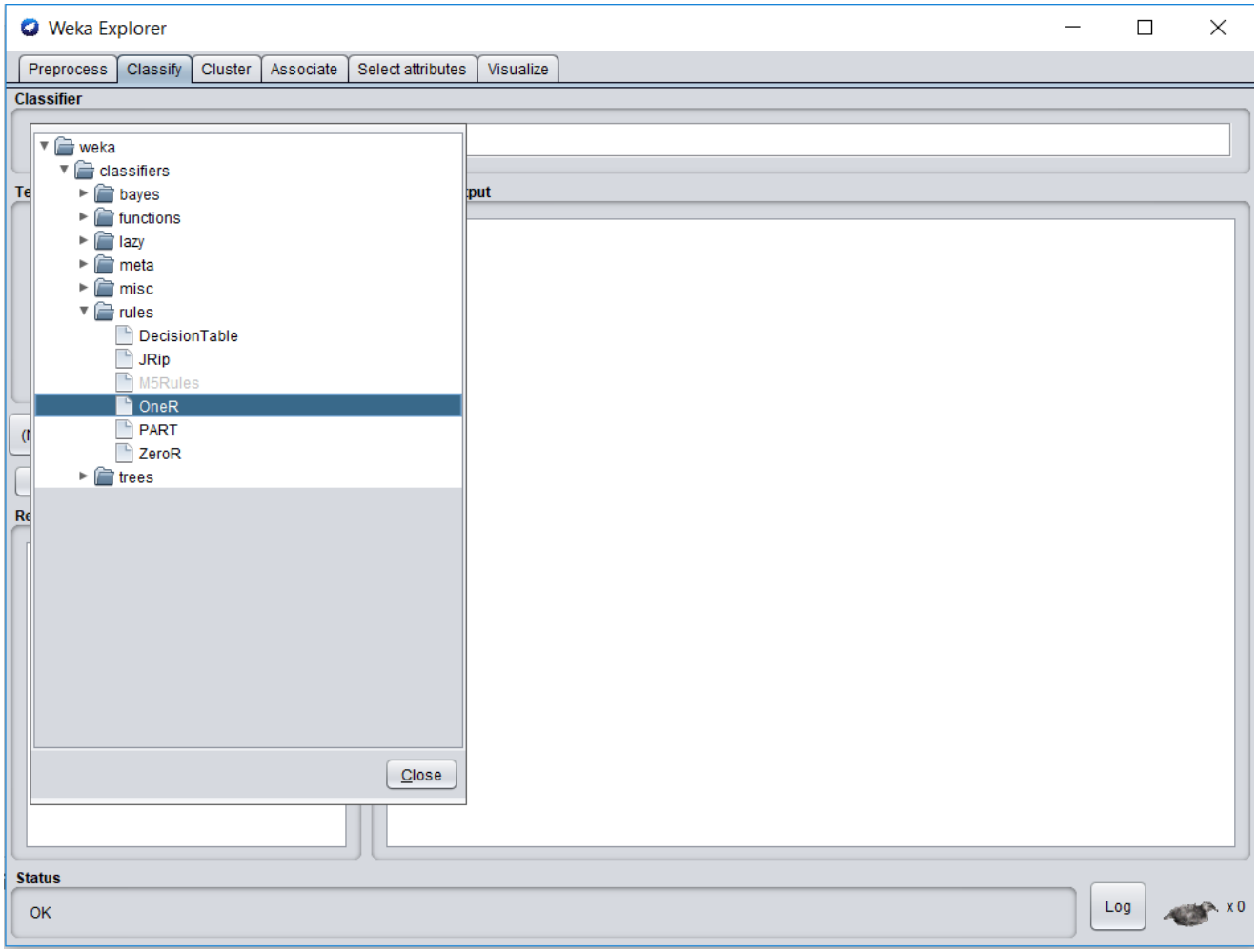
★		Play Golf	
		Yes	No
Outlook	Sunny	3	2
	Overcast	4	0
	Rainy	2	3

IF Outlook = Sunny THEN PlayGolf = Yes
IF Outlook = Overcast THEN PlayGolf = Yes
IF Outlook = Rainy THEN PlayGolf = No

V. Modele de Machine Learning

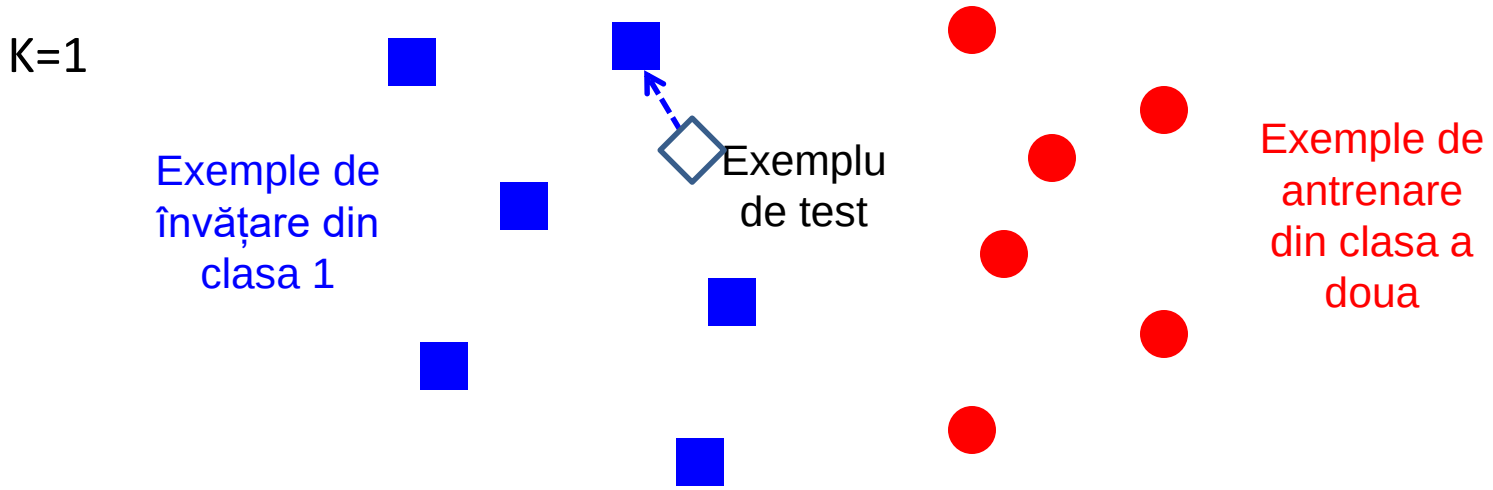
OneR

În imagine se poate vizualiza opțiunea de aplicare a algoritmului OneR în Weka.



V. Modele de Machine Learning

K-Nearest Neighbor
(Cei mai apropiaţi vecini)

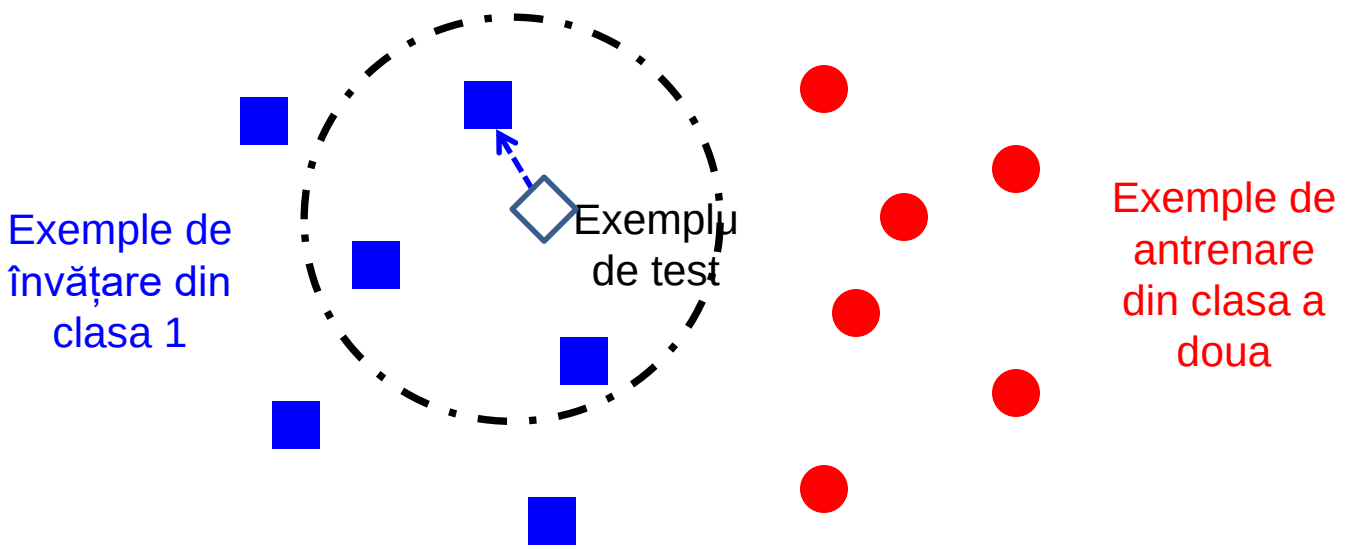


f(**x**) = va lua clasa celui mai apropiat vecin a lui **x**

V. Modele de Machine Learning

K-Nearest Neighbor
(Cei mai apropiaţi vecini)

K=3



V. Modele de Machine Learning

K-Nearest Neighbor
(Cei mai apropiaţi vecini)

Avantaje

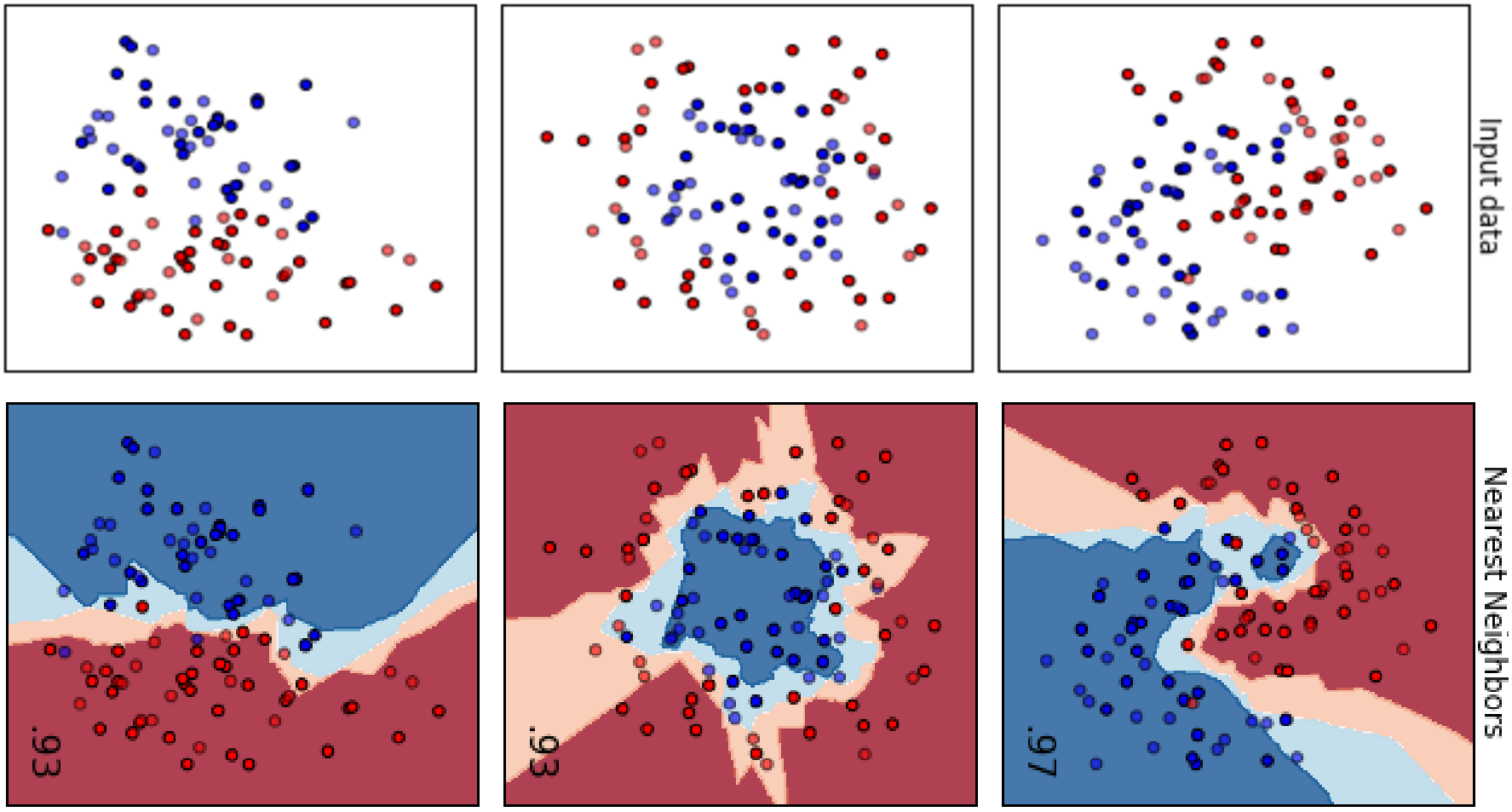
- Simplu (ușor de implementat pentru o primă încercare);
- Nu are nevoie de multe date de învățare

Dezavantaje

- Ocupă foarte multă memorie (se reține în memorie toată baza de antrenare);
- Lent – este nevoie să se compare elementul de clasificat cu toate trăsăturile din baza de date;
- Erori mari de clasificare (nu are rezultate bune pentru probleme complexe);
- Dependent de tipul de metrică utilizat.

V. Modele de Machine Learning

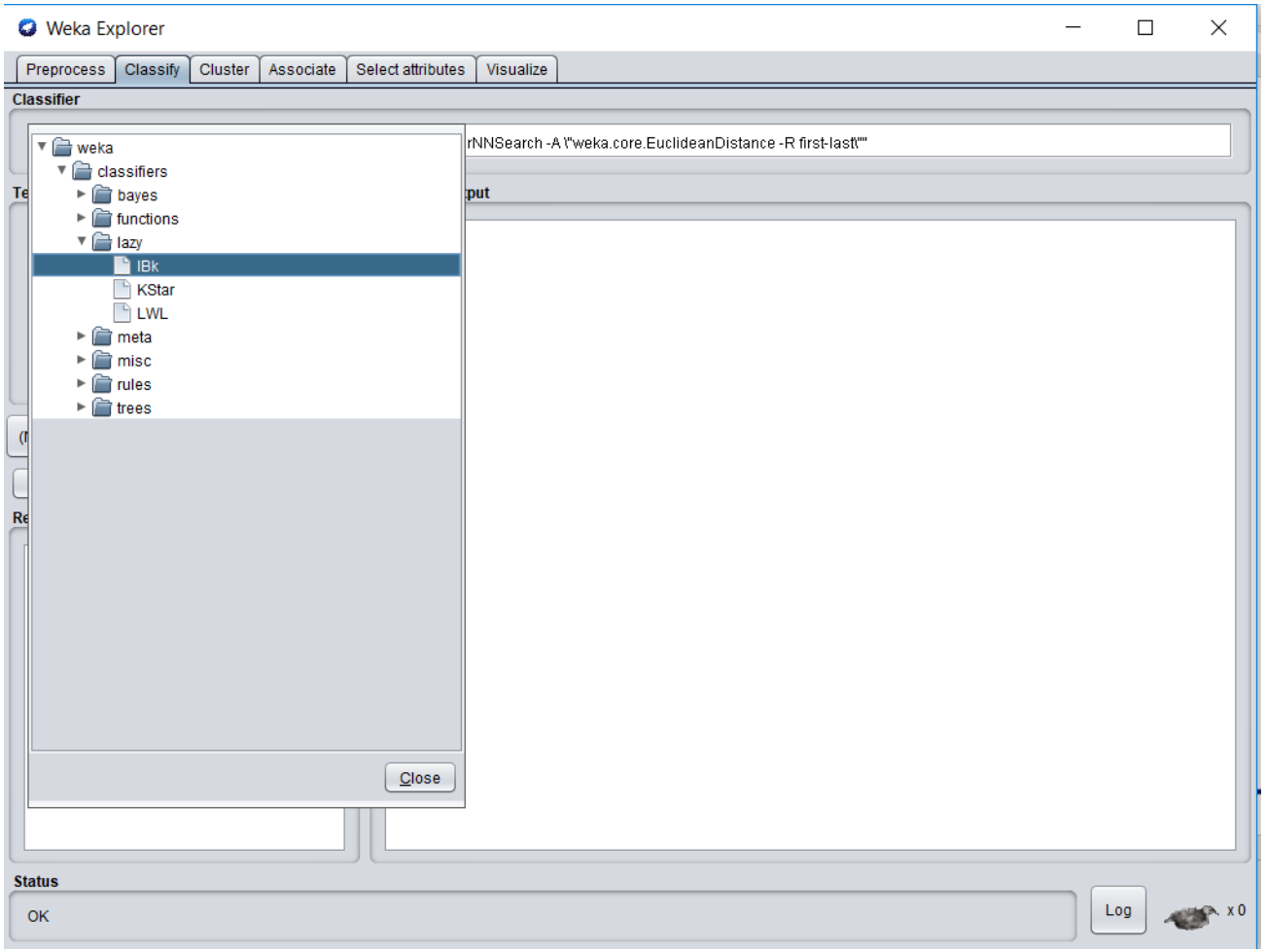
K-Nearest Neighbor
(Cei mai apropiaţi vecini)



V. Modele de Machine Learning

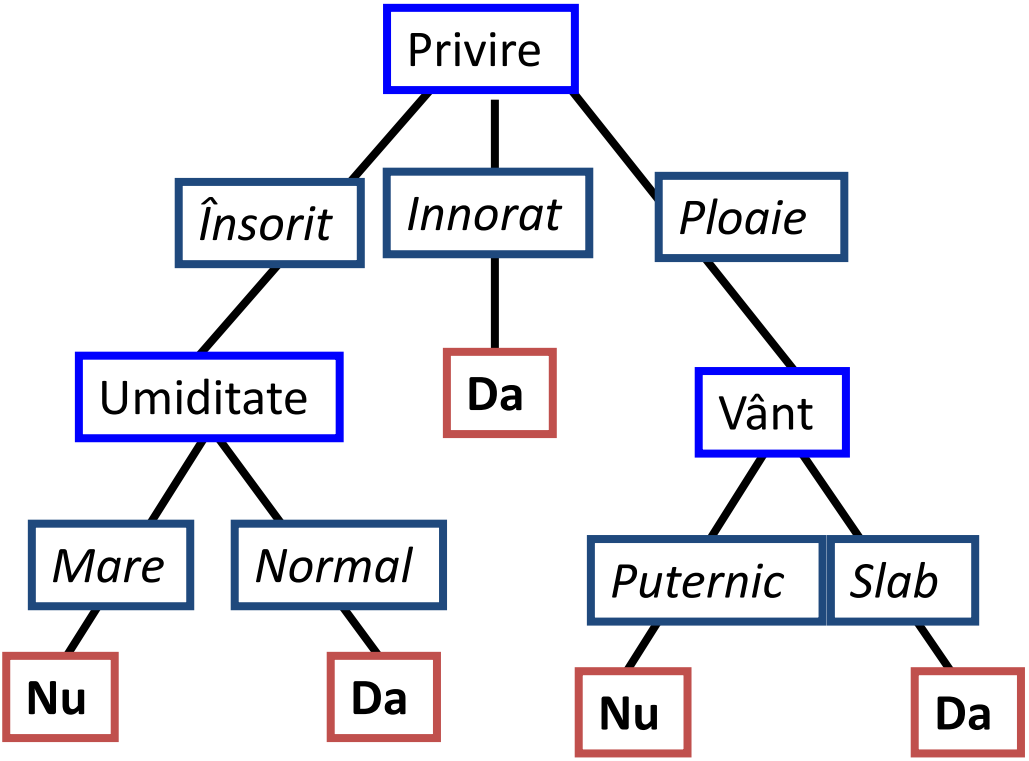
K-Nearest Neighbor

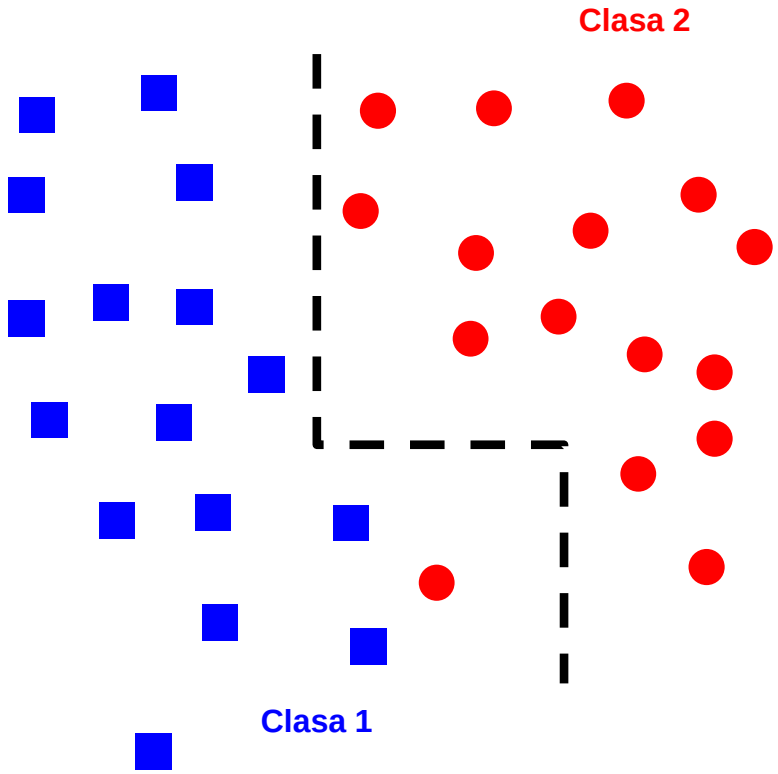
În imagine se poate vizualiza opțiunea de aplicare a algoritmului K-Nearest Neighbor în Weka.



V. Modele de Machine Learning

Arbori de decizie

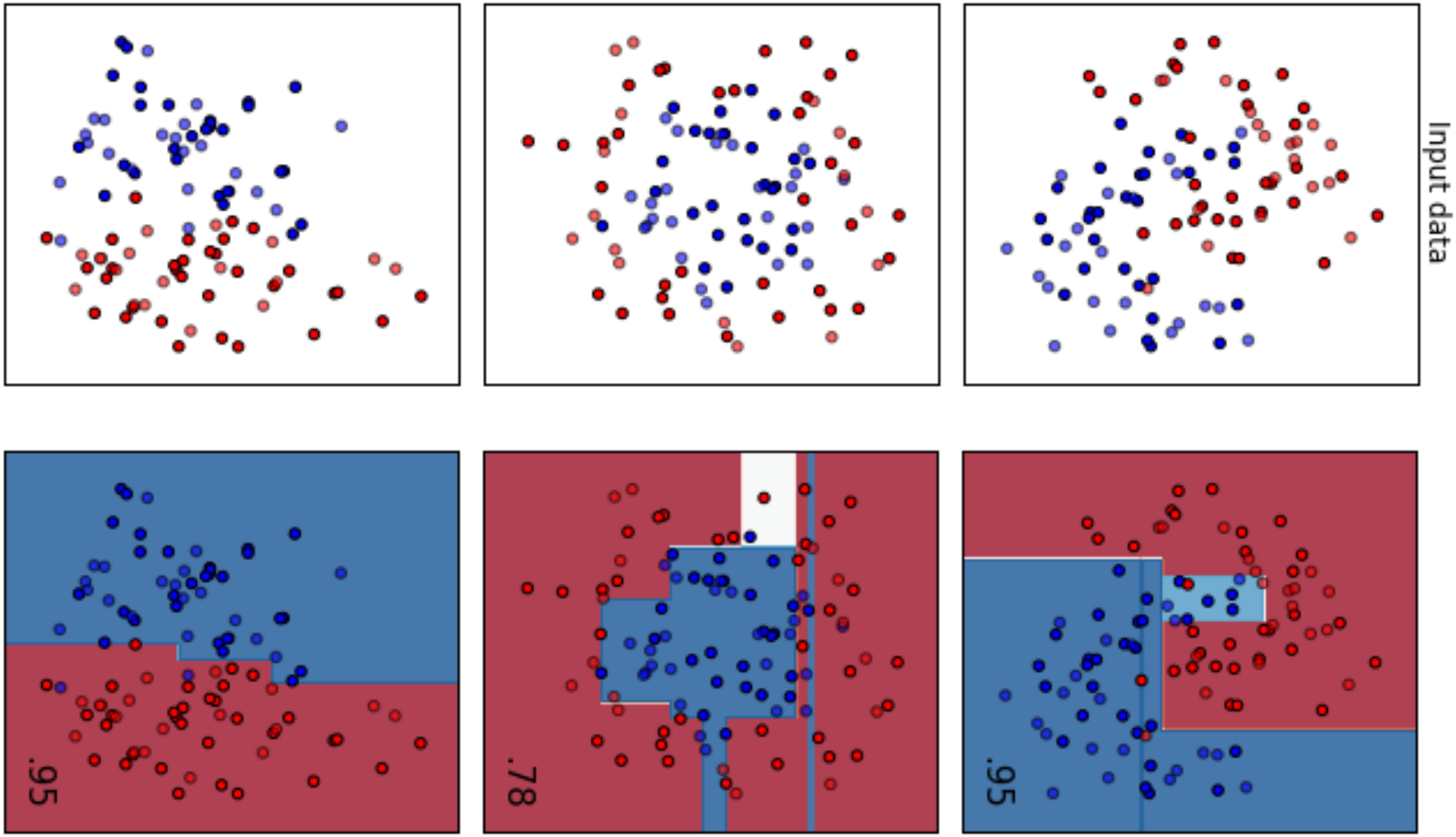




Decizia de a juca tenis

V. Modele de Machine Learning

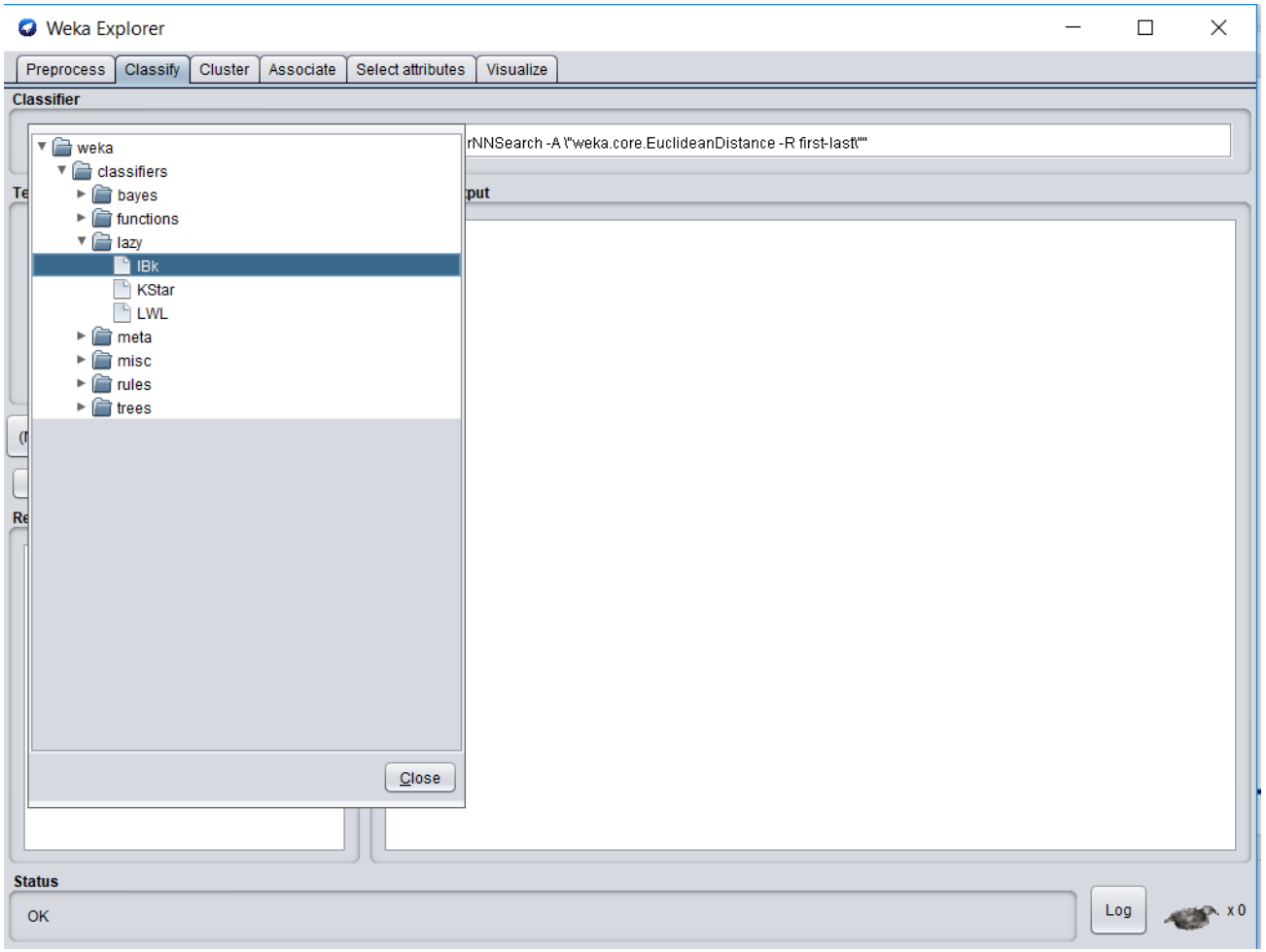
Arbori de decizie



V. Modele de Machine Learning

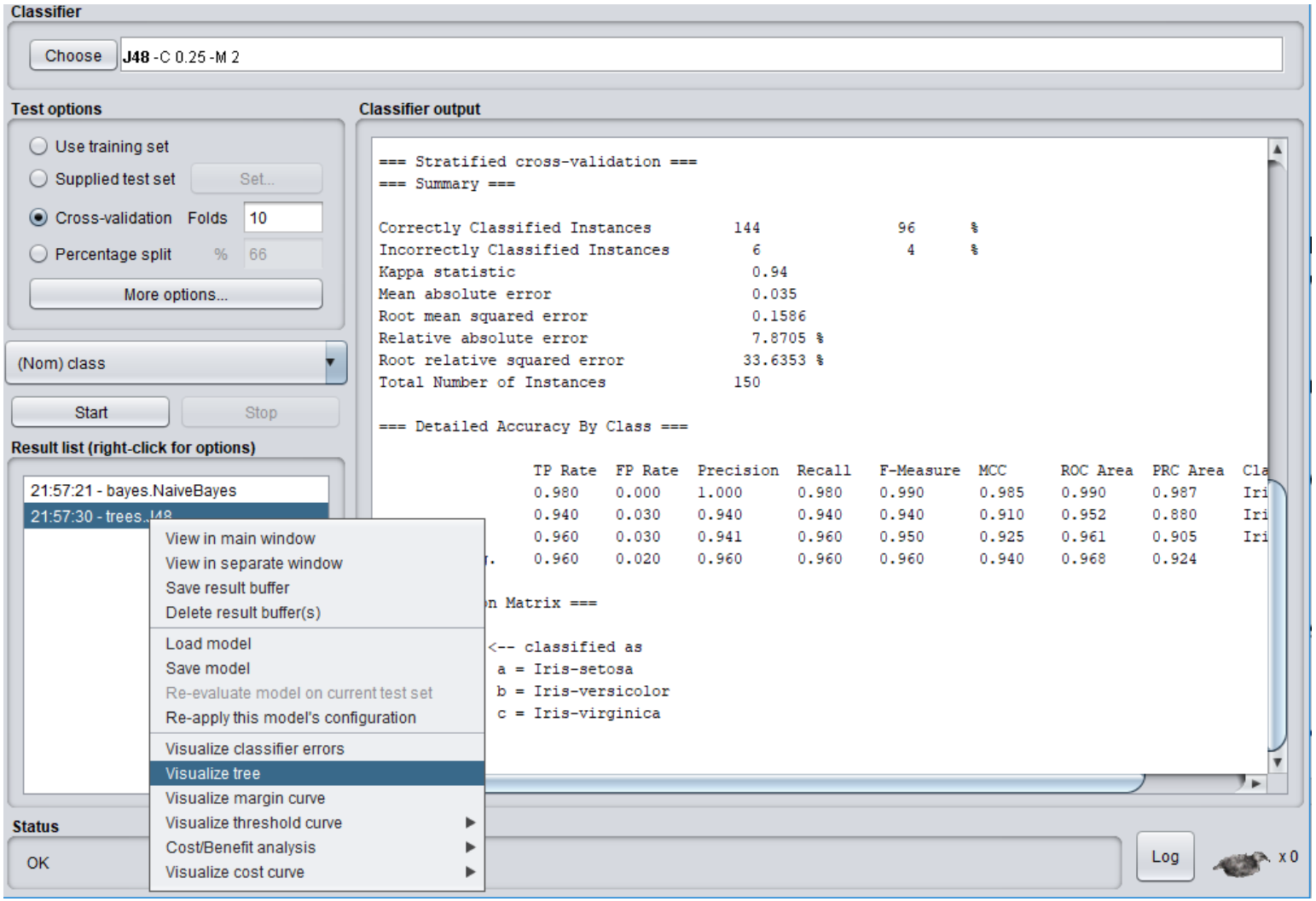
Arbori de decizie

În imagine se poate vizualiza opțiunea de aplicare a algoritmului K-Nearest Neighbor în Weka.



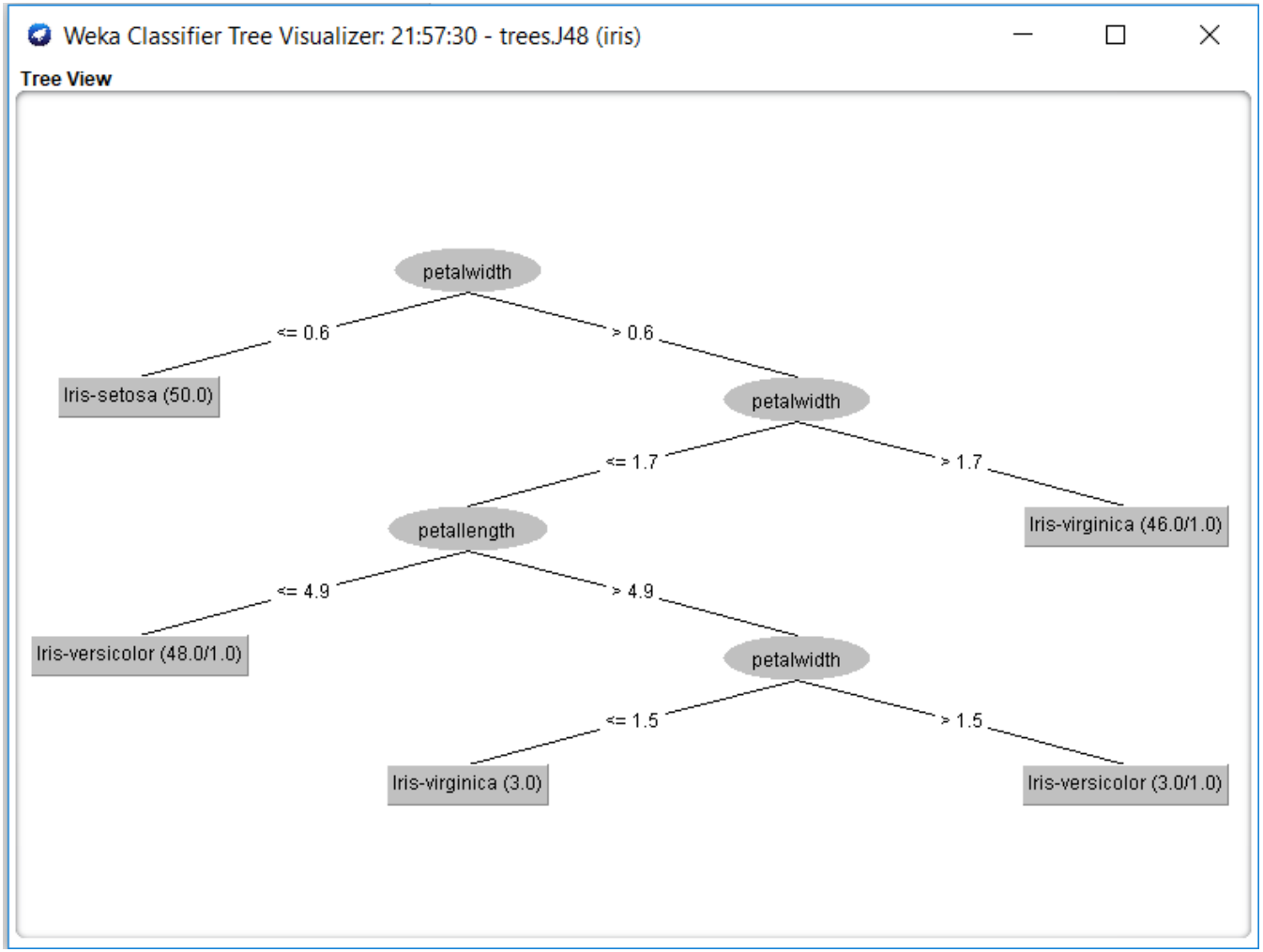
V. Modele de Machine Learning

Vizualizare arbore de decizie



V. Modele de Machine Learning

Vizualizare arbore de decizie



V. Modele de Machine Learning

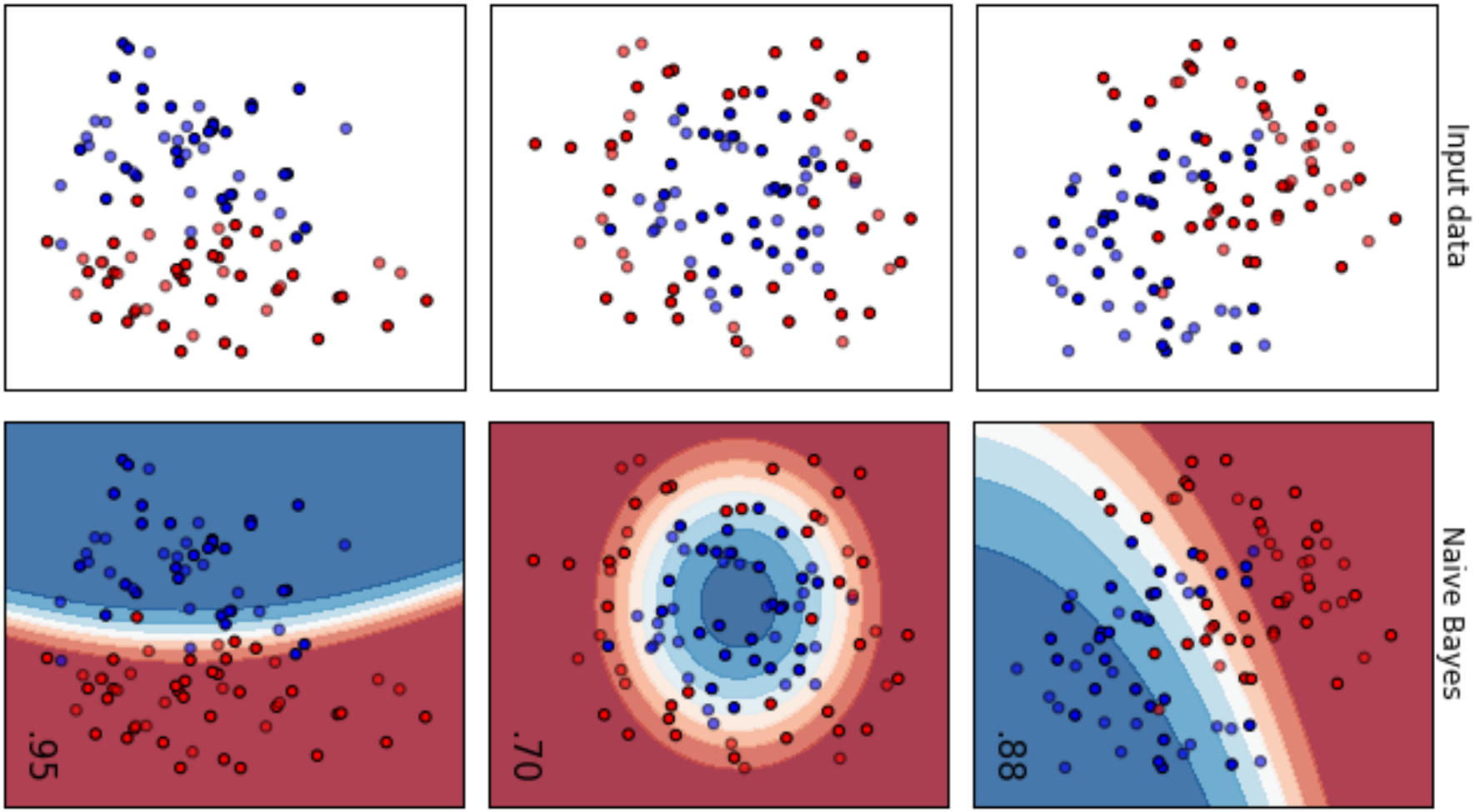
Naive Bayes

Clasificatorul Naive Bayes reprezintă un algoritm probabilistic bazat pe teorema lui Bayes, care pleacă de la premisa de independență între trăsături:

$$P(Y|X_1,\ldots,X_n) = \frac{P(X_1,\ldots,X_n|Y)P(Y)}{P(X_1,\ldots,X_n)}$$

V. Modele de Machine Learning

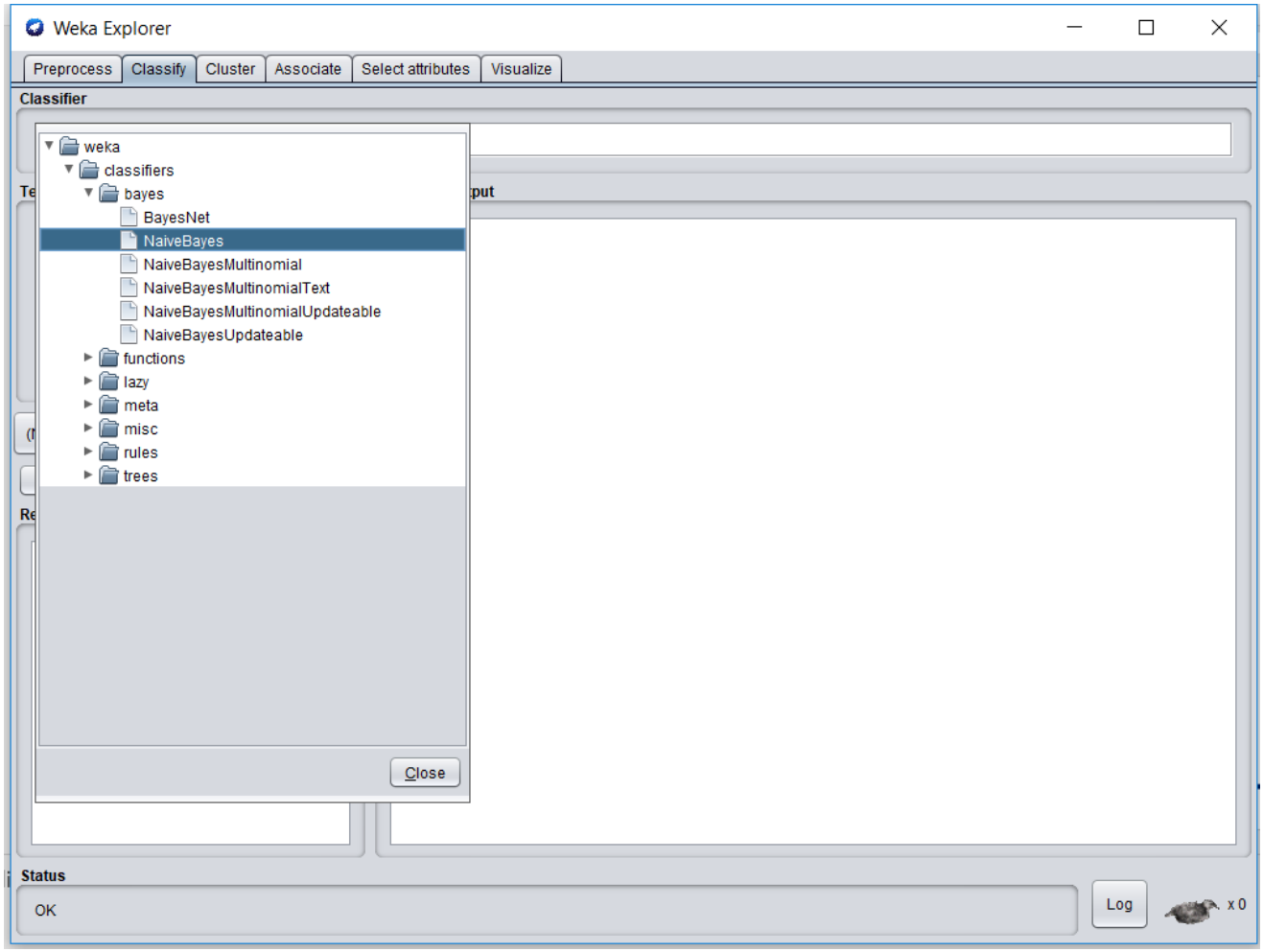
Naive Bayes



V. Modele de Machine Learning

Naive Bayes

În imagine se poate vizualiza opțiunea de aplicare a algoritmului Naive Bayes în Weka.



VI. Exerciții

- Încărcați una din bazele de date:

Weather.arff, *Iris.arff* și *Glass.arff*;

- Utilizați ca și clasificatori: ZeroR, OneR, arbori de decizie (J48), Nearest Neighbor (IBK) și Naive Bayes. Verificați care este cel mai bun clasificator:
  - Rețineți acuratețea de clasificare a fiecărui clasificator în parte;
  - Precizați care este cel mai bun clasificator pentru fiecare problemă în parte.

- Eliminați rând pe rând câte o coloană din bazele de date și comparați acuratețea cu cea obținută anterior;

VI. Exerciții

- Încărcați baza: *dataset_31_credit-g.arff*
- Aplicati posibili algoritmi de preprocesare (prelucrare valori nominale, normalizare)
- Utilizați ca și clasificatori: ZeroR, OneR, arbori de decizie (J48), Nearest Neighbor (IBK) și Naive Bayes. Verificați care este cel mai bun clasificator:
  - Rețineți acuratețea de clasificare a fiecărui clasificator în parte;
  - Precizați care este cel mai bun clasificator pentru fiecare clasă în parte.

Spor!