Systemy uczące się
Ocena zdolności predykcyjnej klasyfikatorów
wykład 5

Jerzy Stefanowski
Instytut Informatyki PP
2021

Akademia Innowacyjnych Zastosowań Technologii Cyfrowych (AI-TECH)
projekt finansowany z środków Programu Operacyjnego Polska Cyfrowa
POPC.03.02.00-00-0001/20

Fundusze Europejskie
Polska Cyfrowa

Rzeczpospolita Polska

Unia Europejska
Europejski Fundusz Rozwoju Regionalnego

Ocena wiedzy klasyfikacyjnej oraz klasyfikatorów

1. Perspektywy oceny klasyfikacji / regresji
2. Miary oceny zdolności predykcyjnych
   - Miary punktowe
   - Miary ROC
   - Uczenie się z kosztami pomyłek
3. Eksperymentalna ocena klasyfikatorów
4. Porównanie wielu klasyfikatorów w studiach przypadków − wykorzystanie testów statystycznych

Różne perspektywy wiedzy klasyfikacyjnej

- Wiedza / klasyfikatory odkryte z danych
  - Predykcja (klasyfikacji) – przewidywanie przydziału nowych obiektów do klas / wykorzystanie jako tzw. klasyfikator (ocena zdolności klasyfikacyjnej – na ogół jedno wybrane kryterium).
  - Opis klasyfikacji obiektów – wyszukiwanie wzorców charakteryzujących właściwości danych i prezentacja ich użytkownikowi w zrozumiałej formie (ocena trudniejsza i bardziej subiektywna) − typowe dla tzw. data mining.

Spójrz też do książki : J.Stefanowski Algorytmy indukcji regul decyzyjnych w odkrywaniu wiedzy 2001. pdf dostępny na mojej stronie WWW

Dlaczego oceniać klasyfikatory?

- Wyzwania praktyczne − potrzeba predykcji
  - Patrz przykłady laboratorium i pierwszy wykład
- Prowadzą do skupienia działania wokół precyzyjnego celu i wspierają decyzje, co do zastosowania
- Pozwalają na porównanie (obecne działanie vs. tzw. baseline; aktualne działania vs. oczekiwane – optymalizacja; porównywanie wielu alternatywnych rozwiązań, …)
- Wspierają tzw. monitoring lub badanie skuteczności systemu
- oraz ….

Tworzenie i ocena klasyfikatorów

Jest procesem trzyetapowym:

1. Konstrukcja modelu w oparciu o zbiór danych wejściowych (przykłady uczące - etykietowane).

   Przykładowe modele :

   - drzewa decyzyjne, reguły (IF .. THEN ..),
   - Naive Bayes, regresja logistyczna,
   - sieci neuronowe, SVM, zespoły.

2. Ocena modelu (przykłady testujące − ukryte etykiety)
3. Użycie/ wdrożenie modelu (klasyfikowanie nowych obiektów − bez etykiet)

Popularne kryteria

- **Trafność predykcji** (klasyfikacja / regresja)
- Zdolności interpretacji modelu: np. drzewa decyzyjne vs. sieci neuronowe => patrz dalsze wykłady
- Złożoność struktury, np.
  - rozmiar drzew decyzyjnego,
  - miary oceny reguły
- Odporność na różne charakterystyki danych
  - Szum (noise),
  - Inne trudności rozkładu danych,

------- oraz wymagania obliczeniowe

- Szybkość i skalowalność:
  - czas uczenia się,
  - szybkość samego klasyfikowania

Trafność klasyfikowania

- Użyj przykładów testowych nie wykorzystanych w fazie uczenia klasyfikatora:
  - N_t – liczba przykładów testowych
  - N_c – liczba poprawnie sklasyfikowanych przykładów testowych
- Trafność klasyfikowania (ang. classification accuracy) − najczęściej wyrażania w procentach:

$$\eta = \frac{N_c}{N_t}$$

- Alternatywnie błąd klasyfikowania. $\varepsilon = \frac{N_t - N_c}{N_t}$

Pomyśl – czy oba błędy się zawsze uzupełniają (np. do 1,0 lub 100%)?

Predykcja zmiennej *y* (liczbowej)

- Zmienna wyjściowa liczbowa : ocena jak odbiega predykcja y^ od właściwej wyjściowej y
- Odpowiednik funkcji straty : ocena różnicy y_i oraz predykcji y_i^
  - Absolute error: $| y_i - y^_i |$
  - Squared error: $(y_i - y^_i)^2$
- Popularne uśrednione wartości błędów
  - Mean absolute error: $\frac{\sum_{i=1}^{n} | y_i - \hat{y}_i |}{n}$ **Mean squared error**: $\frac{\sum_{i=1}^{n} (y_i - \hat{y}_i)^2}{n}$
  - Relative absolute error: $\frac{\sum_{i=1}^{n} | y_i - \hat{y}_i |}{\sum_{i=1}^{n} | y_i - \bar{y} |}$ Relative squared error: $\frac{\sum_{i=1}^{n} (y_i - \hat{y}_i)^2}{\sum_{i=1}^{n} (y_i - \bar{y})^2}$
  - Na ogół stosowane (square) root mean-square error, oraz root relative squared error

Wiele innych miar oceny predykcji

`roc_curve` (y_true, y_score[, pos_label, ...])	Compute Receiver operating characteristic (ROC)
`balanced_accuracy_score` (y_true, y_pred[, ...])	Compute the balanced accuracy

Others also work in the multiclass case:

`cohen_kappa_score` (y1, y2[, labels, weights, ...])	Cohen's kappa: a statistic that measures inter-annotator agreement.
`confusion_matrix` (y_true, y_pred[, labels, ...])	Compute confusion matrix to evaluate the accuracy of a classification
`hinge_loss` (y_true, pred_decision[, labels, ...])	Average hinge loss (non-regularized)
`matthews_corrcoef` (y_true, y_pred[, ...])	Compute the Matthews correlation coefficient (MCC)

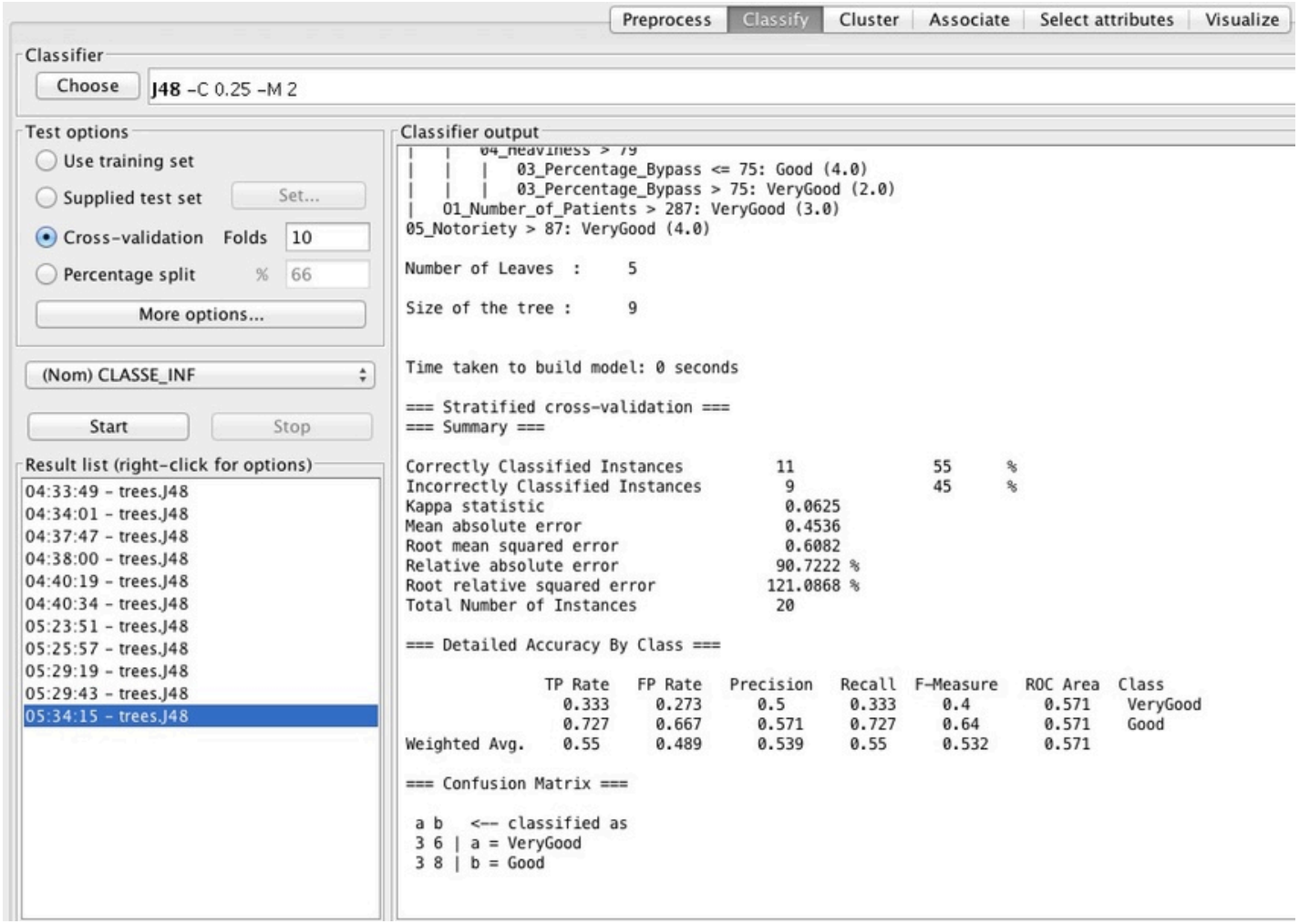
Some also work in the multilabel case:

`accuracy_score` (y_true, y_pred[, normalize, ...])	Accuracy classification score.
`classification_report` (y_true, y_pred[, ...])	Build a text report showing the main classification metrics
`f1_score` (y_true, y_pred[, labels, ...])	Compute the F1 score, also known as balanced F-score or F-measure
`fbeta_score` (y_true, y_pred, beta[, labels, ...])	Compute the F-beta score
`hamming_loss` (y_true, y_pred[, labels, ...])	Compute the average Hamming loss.
`jaccard_similarity_score` (y_true, y_pred[, ...])	Jaccard similarity coefficient score
`log_loss` (y_true, y_pred[, eps, normalize, ...])	Log loss, aka logistic loss or cross-entropy loss.
`precision_recall_fscore_support` (y_true, y_pred)	Compute precision, recall, F-measure and support for each class
`precision_score` (y_true, y_pred[, labels, ...])	Compute the precision
`recall_score` (y_true, y_pred[, labels, ...])	Compute the recall
`zero_one_loss` (y_true, y_pred[, normalize, ...])	Zero-one classification loss.

And some work with binary and multilabel (but not multiclass) problems:

Za dokumentacją scikit learn

WEKA evaluation



Źródło − własne uruchomienie oprogramowania

Miary – zależność od zadania

- Klasyfikacja binarna
  - Wskazanie etykiety vs. scoring predictions
    - Miary punktowe np. Accuracy,
    - Ocena prawdopodobieństwa − Kappa statistics
    - Zainteresowanie wybraną klasą
      - Precision, Recall / Sensitivity, Specificity, F-score, G-mean
  - Miary graficzne: ROC, PRcurves, Lift curves
- Wieloklasowość / Wielo-etykietowość
  - Nie wszystkie miary binarne można uogólnić
- Specyfika danych
  - Tzw. Imbalanced data oraz cost sensitive learning
- Predykcja ciągła
  - Błędy RSME, oceny różnic rozkładów (dywergencje KL)

Macierz pomyłek

- Analiza pomyłek w przydziale do różnych klas przy pomocy tzw. macierz pomyłek (ang. *confusion matrix*)
- Macierz *r*×*r*, gdzie wiersze odpowiadają poprawnym klasom decyzyjnym, a kolumny decyzjom przewidywanym przez klasyfikator; na przecięciu wiersza *i* oraz kolumny *j* - liczba przykładów *n-ij* należących oryginalnie do klasy *i*-tej, a zaliczonej do klasy *j*-tej

Przykład:

	Przewidywane klasy decyzyjne		
Oryginalne klasy	K_1	K_2	K_3
K_1	50	0	0
K_2	0	48	2
K_3	0	4	46

Klasyfikacja binarna

- Niektóre zastosowania → jedna z klas posiada szczególne znaczenie, np. diagnozowanie poważnej choroby. Zadanie → klasyfikacja binarna.

Oryginalne klasy	Przewidywane klasy decyzyjne	
	Pozytywna	Negatywna
Pozytywna	*TP*	*FN*
Negatywna	*FP*	*TN*

- Nazewnictwo (inspirowane medycznie):
  - *TP* (ang. *true positive*) – liczba poprawnie sklasyfikowanych przykładów z wybranej klasy (ang. *hit*),
  - *FN* (ang. *false negative*) – liczba błędnie sklasyfikowanych przykładów z tej klasy, tj. decyzja negatywna podczas gdy przykład w rzeczywistości jest pozytywny (błąd pominięcia - z ang. *miss*),
  - *TN* (ang. *true negative*) – liczba przykładów poprawnie nie przydzielonych do wybranej klasy (poprawnie odrzuconych z ang. *correct rejection*),
  - *FP* (ang. *false positive*) – liczba przykładów błędnie przydzielonych do wybranej klasy, podczas gdy w rzeczywistości do niej nie należą (ang. *false alarm*).

Trudności oceny trafności

Oryginalna→	Pos	Neg
Yes	200	100
No	300	400
	P=500	N=500

Oryginalna →	Pos	Neg
Yes	400	300
No	100	200
	P=500	N=500

Oba klasyfikatory = 60% trafność (accuracy)

Lecz różnie w predykcji poszczególnych klas:

- Lewa tabela: niski TPR /wysoka rozpoznawalnośc Neg
- Prawa tabela: dobre rozpoznawanie klasy Pos, słabe Neg

Zainteresowanie pojedyncza klasą

- **Dane niezbalansowane** (na ogół dwie klasy)
  - (ang. imbalanced data) klasy nie są w przybliżeniu równo liczne; Klasa mniejszościowa (ang. minority class) zawiera wyraźnie mniej przykładów niż inne klasy
  - Przykłady z klasy mniejszościowej są często najważniejsze i ich poprawne rozpoznawanie jest głównym celem.
    - Rozpoznawanie rzadkiej, niebezpiecznej choroby
- Powoduje trudności w fazie uczenia i obniża zdolność predykcyjną
  - Niektóre klasyfikatory pomimo wysokiej globalnej trafności nie rozpoznają kl. mniejszościowej
  - Przykład klasyfikacji tekstów (Catlett) trafność 99% , lecz brak rozpoznania specjalnych dokumentów (TPR 0%)

Miary punktowe dla niezbalansowania klas

Rozpoznawanie klasy mniejszościowej z …

Wiele miar definiowany na podstawie macierzy pomyłek

$$Sensitivity = Recall = \frac{TP}{TP + FN}$$

$$Specificity = \frac{TN}{TN + FP} \qquad Precision = \frac{TP}{TP + FP}$$

Oryginalne	Przewidywane	
	+	-
+	*TP*	*FN*
-	*FP*	*TN*

Inne miary:

False-positive rate = *FP* / (*FP*+*TN*), czyli 1 – specyficzność

Agregacje: $G\text{-}mean = \sqrt{Sensitivity * Specificity}$

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

Analiza macierzy… spróbuj rozwiązać…

$$Sensitivity = \frac{TP}{TP+FN} = ?$$

$$Specificity = \frac{TN}{TN+FP} = ?$$

Co przewidywano

Rzeczywista Klasa	**1**	**0**
1	60	30
0	80	20

60+30 = 90 przykładów w danych należało do Klasy 1

80+20 = 100 przykładów było w Klasy 0

90+100 = 190 łączna liczba przykładów

Który klasyfikator jest najlepszy – miary mogą oceniać inne aspekt, np. eksperymenty UCI Breast Cancer

Algo	Acc	RMSE	TPR	FPR	Prec	Rec	F	AUC	Info S
NB	71.7	.4534	.44	.16	.53	.44	.48	.7	48.11
C4.5	75.5	.4324	.27	.04	.74	.27	.4	.59	34.28
3NN	72.4	.5101	.32	.1	.56	.32	.41	.63	43.37
Ripper	71	.4494	.37	.14	.52	.37	.43	.6	22.34
SVM	69.6	.5515	.33	.15	.48	.33	.39	.59	54.89
Bagg	67.8	.4518	.17	.1	.4	.17	.23	.63	11.30
Boost	70.3	.4329	.42	.18	.5	.42	.46	.7	34.48
RanFR	69.23	.47	.33	.15	.48	.33	.39	.63	20.78

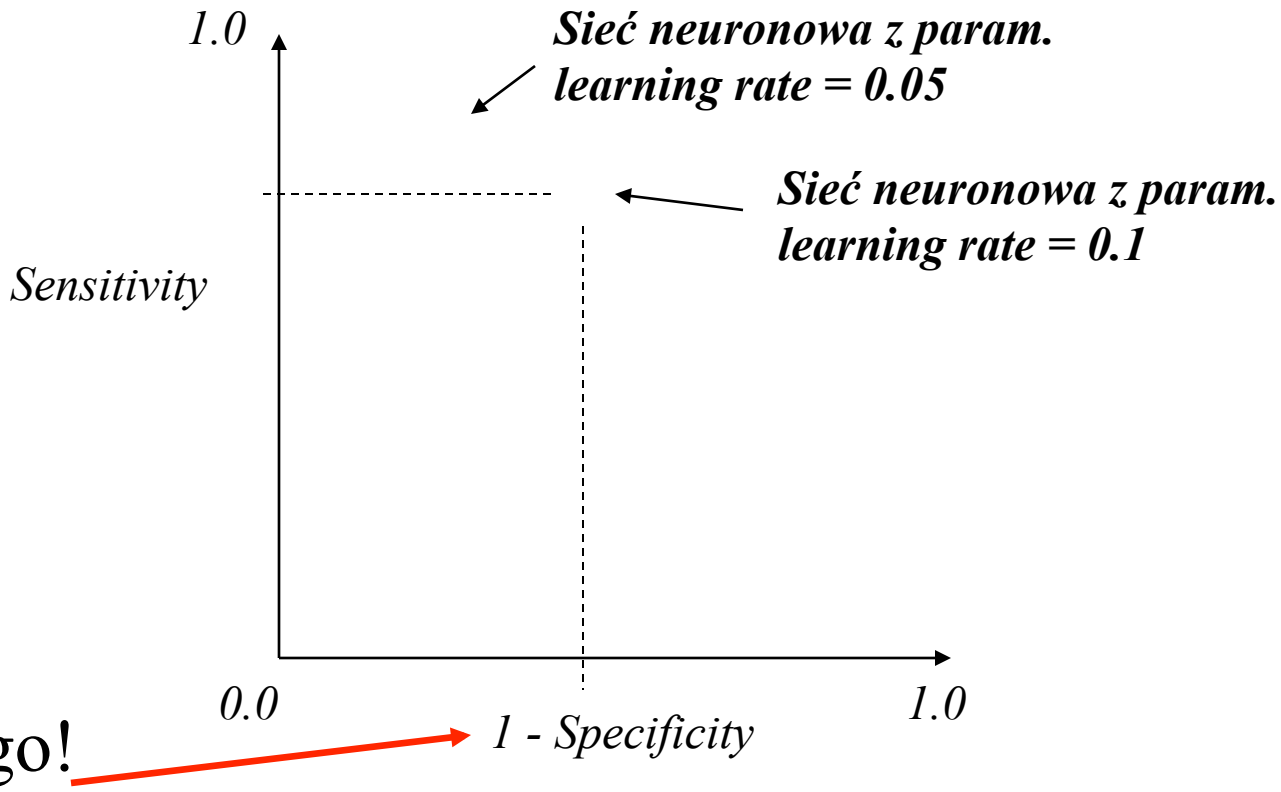
Przykład inspirowany wykładem N.Japkowicz

Scoring classifier – odpowiedź także liczbowa (np. NB, ANN, regresja logistyczna)

- Klasyfikator oprócz wskazania klasy pokazuje także wartość ilościową z nią związaną
  - Pomyśl o Naiwnym klasyfikatorze Bayesowskim
- Ponadto tzw. klasyfikator ciągły- możliwość progowania wyjścia modelu – zwłaszcza dla dwóch klas

Analiza krzywej ROC

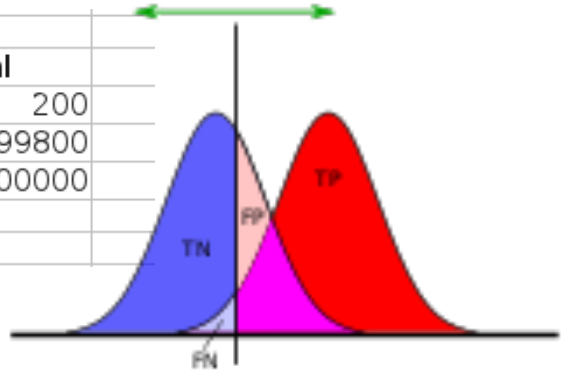
Każda technika budowy klasyfikatora może być scharakteryzowana poprzez pewne wartości miar ‘sensitivity’ i ‘specificity’. Graficznie można je przedstawić na wykresie ‘sensitivity’ vs. 1 – ‘specificity’.



Dlaczego!

Interpretacja progu klasyfiktora

	Test Positive	Test Negative	Total
Patient Diseased	160	40	200
Patient Healthy	29940	69860	99800
Total	30100	69900	100000

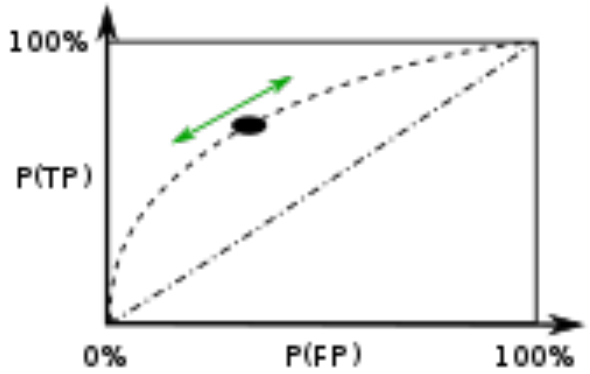


TP	FP
FN	TN

Maximum, np. 1

Próg - T

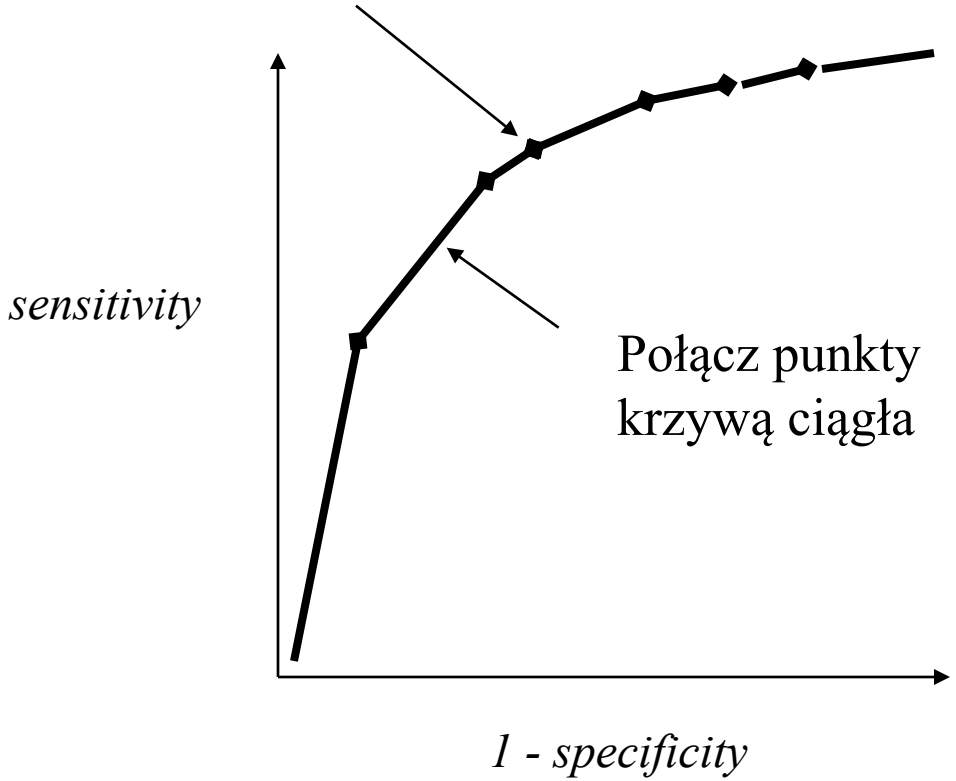
Minimum, np. 0



Źródło - Wikipedia

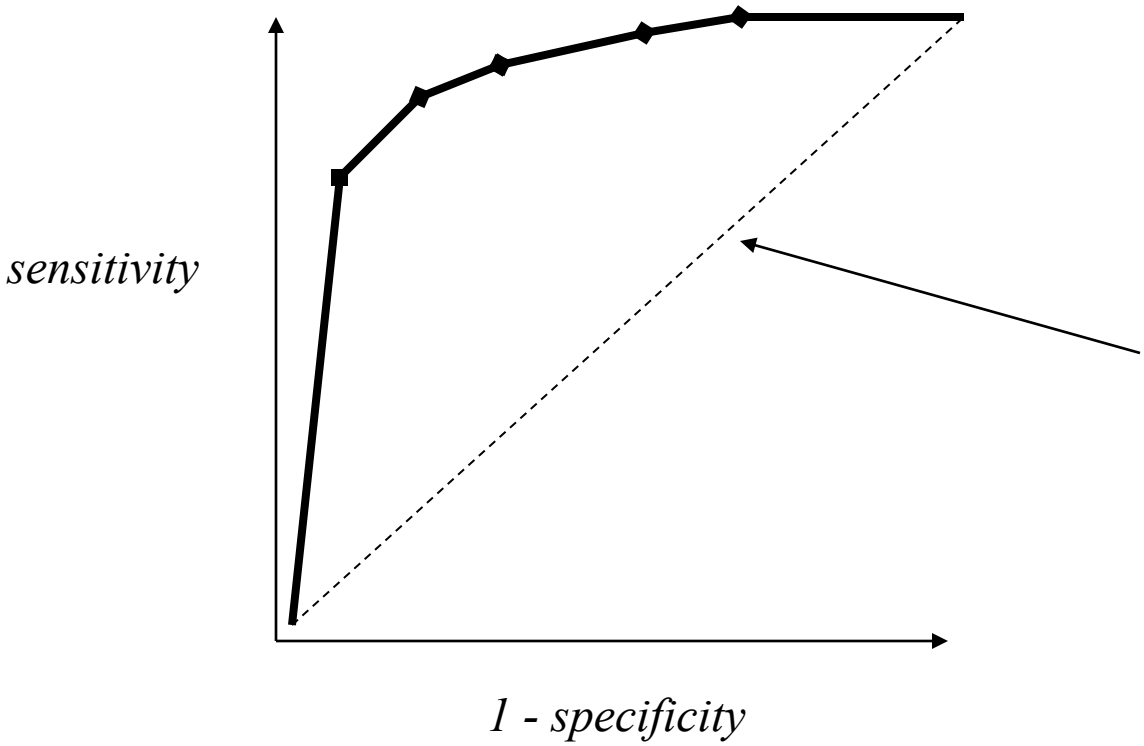
ROC - analiza

Algorytm może być parametryzowany, i w rezultacie otrzymuje się serie punktów odpowiadających doborowi parametrów



Wykres nazywany
‘krzywą’ ROC.

Krzywa ROC

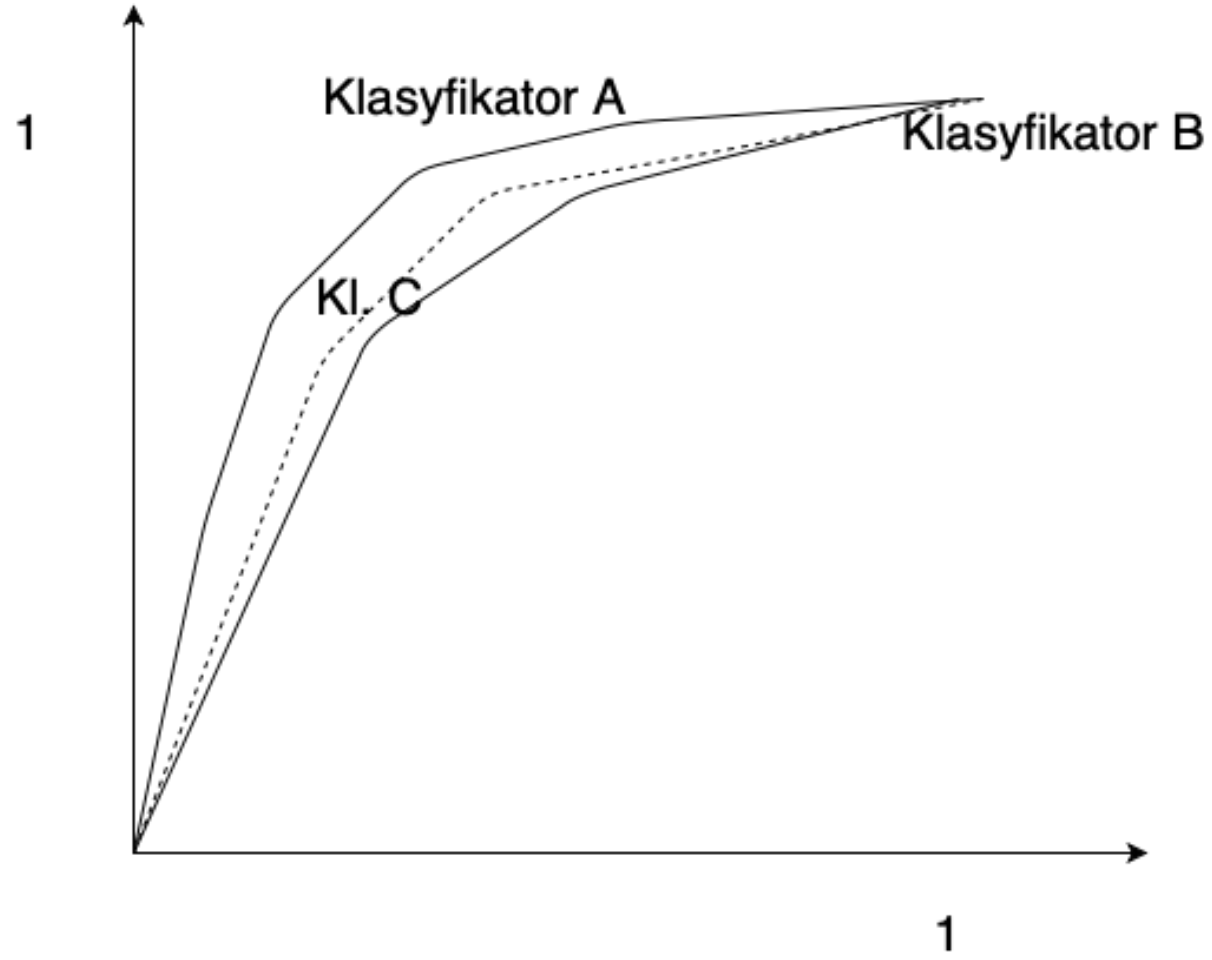


Im krzywa bardziej wygięta ku górnemu lewemu narożnikowi, tym lepszy klasyfikator .

Przekątna odpowiada losowemu „zgadywaniu”. Im bliżej niej, tym gorszy klasyfikator

Można porównywać działanie kilku klasyfikatorów.
Miary oceny np. **AUC** – pole pod krzywą. Wartość z zakresu 0 do 1

Porównywanie działania klasyfikatorów na ROC



Krzywe dla 3 różnych klasyfikatorów − A najlepszy

Krzywe mogę się przecinąć

Macierze kosztów

Oryginalne klasy	Przewidywane klasy decyzyjne	
	Pozytywna	Negatywna
Pozytywna	*C(TP)*	*C(FP)*
Negatywna	*C(FN)*	*C(TN)*

Koszty C(TP) i C(TN) -> 0; a C(FP) na ogół większe niż C(FN)

Oryginalne klasy	Przewidywane klasy decyzyjne	
	Pozytywna	Negatywna
Pozytywna	*0*	*15*
Negatywna	*5*	*0*

Pomyłki mają różną interpretacje i praktyczne znaczenie

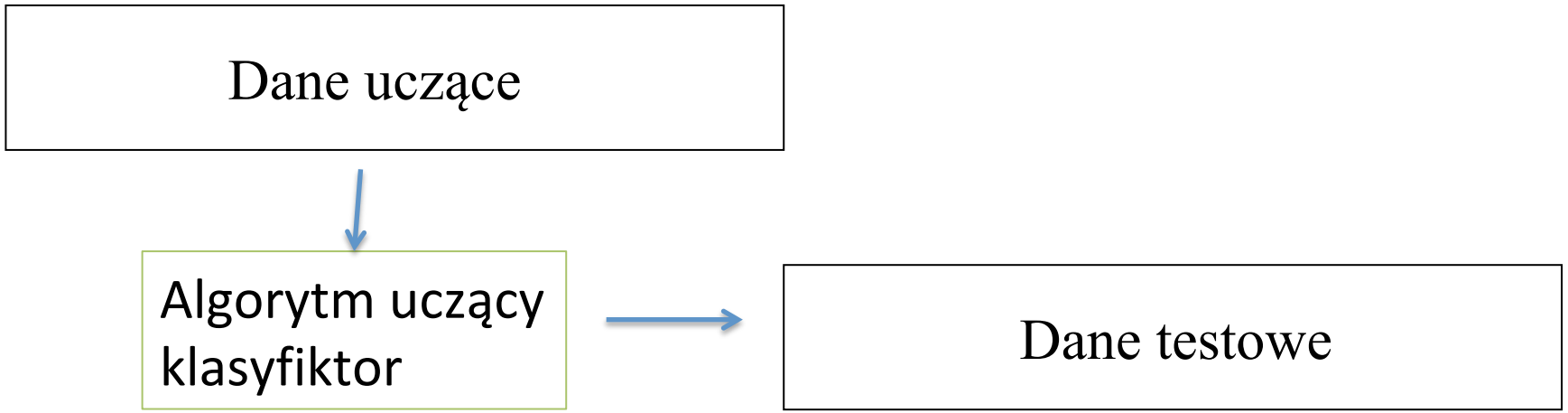
Implementacje: np. WEKA costsensitiveclassifiers

Jak szacować wiarygodnie ?

- Zależy od perspektywy użycia wiedzy:
  - Predykcja klasyfikacji albo opisowa
- Ocena na zbiorze uczącym nie jest wiarygodna jeśli rozważamy predykcję nowych faktów!
  - Nowe obserwacje najprawdopodobniej nie będą takie same jak dane uczące!
  - Choć zasada reprezentatywności próbki uczącej …
- Problem przeuczenia (ang. overfiting)
  - Nadmierne dopasowanie do specyfiki danych uczących powiązane jest najczęściej z utratą zdolności uogólniania (ang. generalization) i predykcji nowych faktów!

Zasada eksperymentalnej oceny

Niezależny zbiór przykładów testowych - nie wykorzystuj w fazie uczenia klasyfikatora!

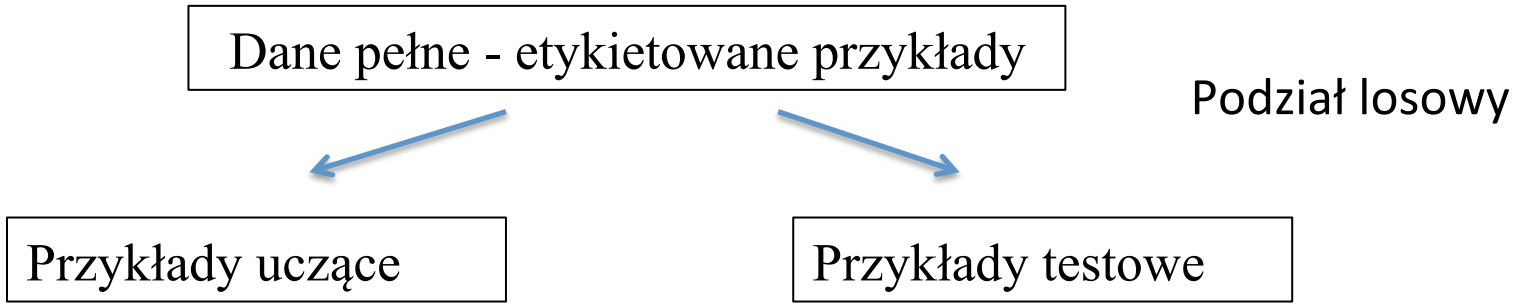


Nie dopuszczaj do tzw. przecieku informacji (ang. information leak)

Błąd treningowy – niebezpieczeństwo przeuczenia.

Podejście empiryczne

- Zasada „Train and test” (ucz i testuj)
- Gdy nie ma podziału zadanego przez nauczyciela, to co wykorzystasz - losowe podziały.
  - **Podziały – próba losowa LECZ ile i jakie przykłady!**
- Nadal pytanie jak szacować wiarygodnie?

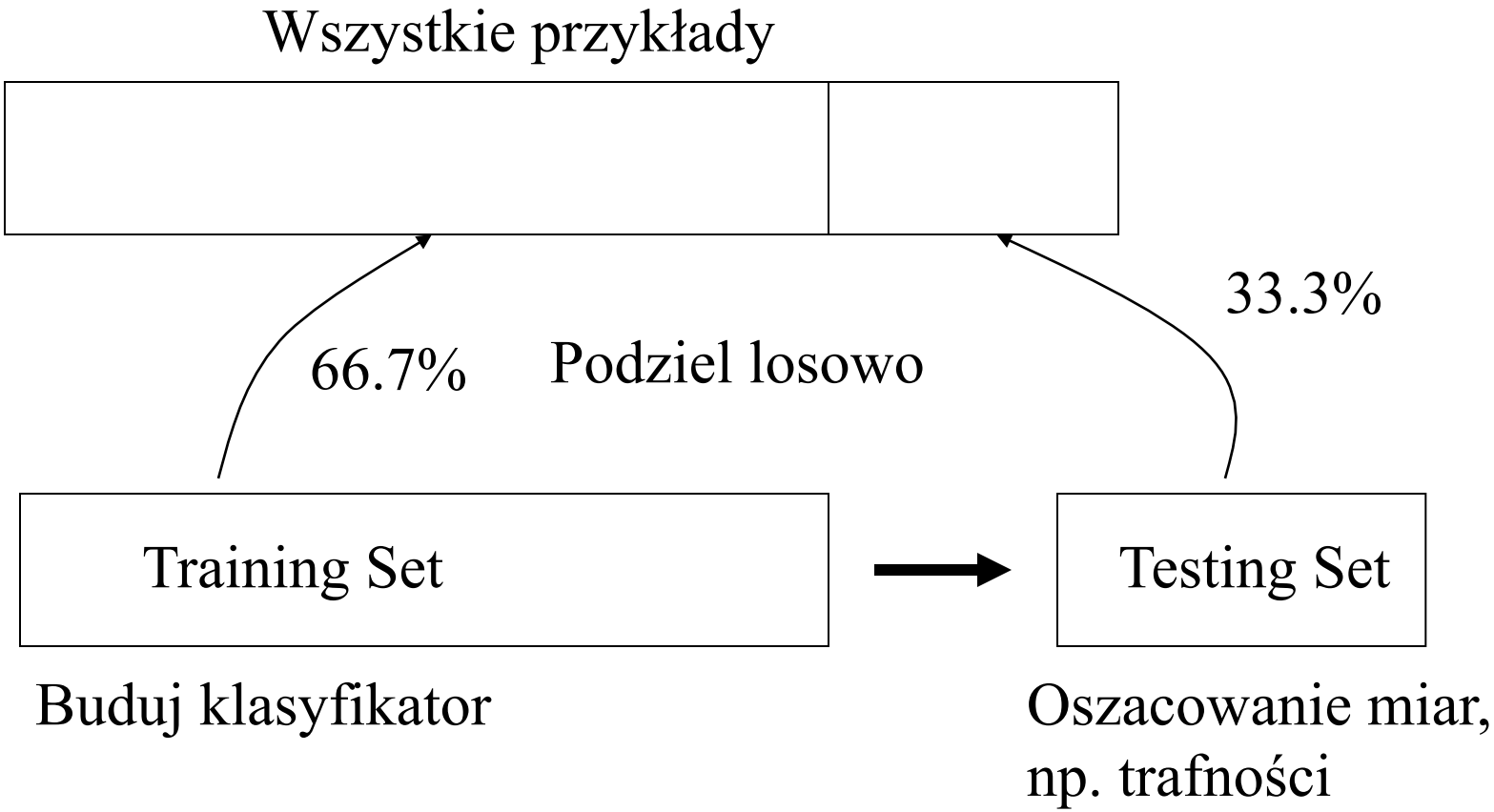


- Typowo − każdy przykład ma równe prawdopodobieństwa wylosowania podziału
- Wersje spec. losowania − zmienne prawdopodobieństwa

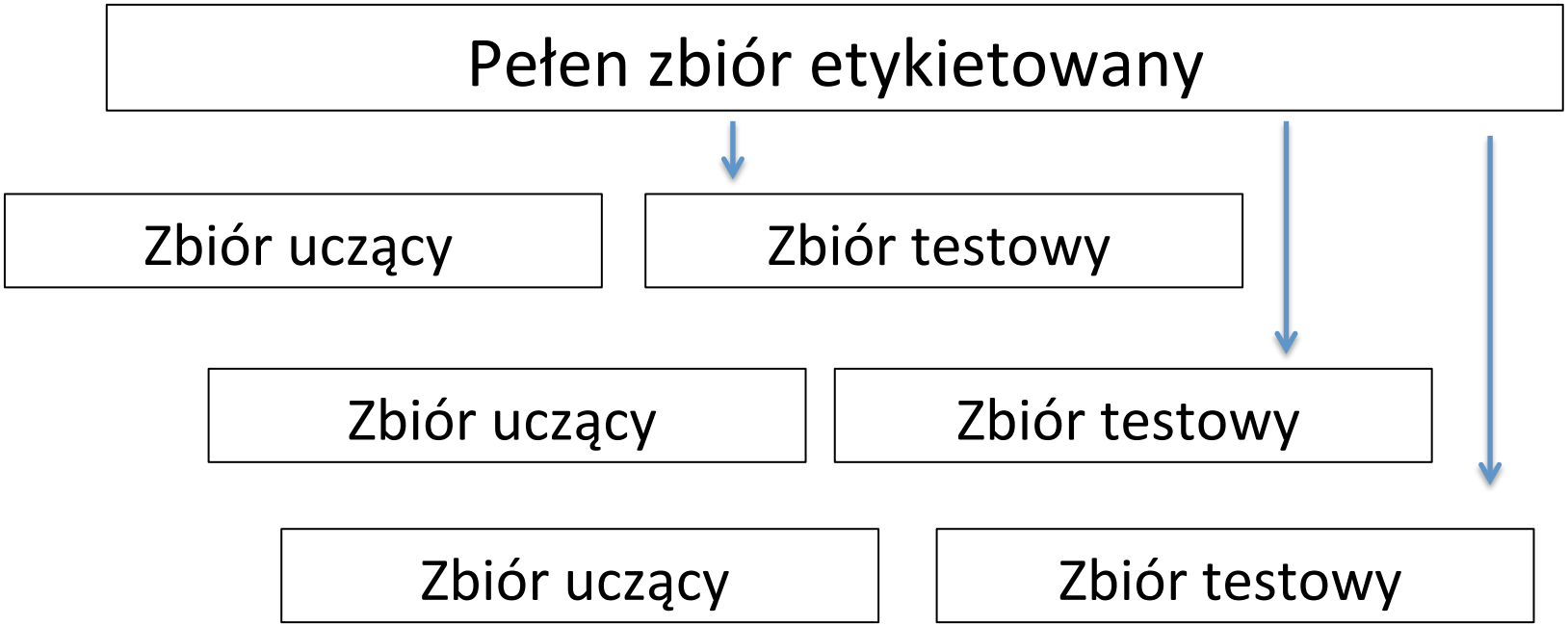
Empiryczne metody estymacji

- Techniki podziału: „**hold-out**” (bardzo duża l. przykładów)
  - Użyj dwóch niezależnych zbiorów: uczącego (2/3), testowego (1/3)
  - Jednokrotny podział losowy stosuje się dla dużych zbiorów (hold-out)
- „**Cross-validation**” - Ocena krzyżowa
  - Podziel losowo dane w *k* podzbiorów (równomierne lub warstwowe)
  - Użyj *k-1* podzbiorów jako części uczącej i pozostałej jako testującej (*k*-fold cross-validation).
  - Oblicz wynik średni.
  - Stosowane dla danych o średnich rozmiarach (najczęściej *k* = 10)
- **leaving-one-out** = Dla małych rozmiarów danych < 100 przykłądów.
  - „Leaving-one-out” jest szczególnym przypadkiem, dla którego liczba iteracji jest równa liczbie przykładów
- Specjalne techniki statystyczne dla mniejszej l. przykładów

Jednokrotny podział (hold-out) – duża liczba przykładów (> tysięcy)



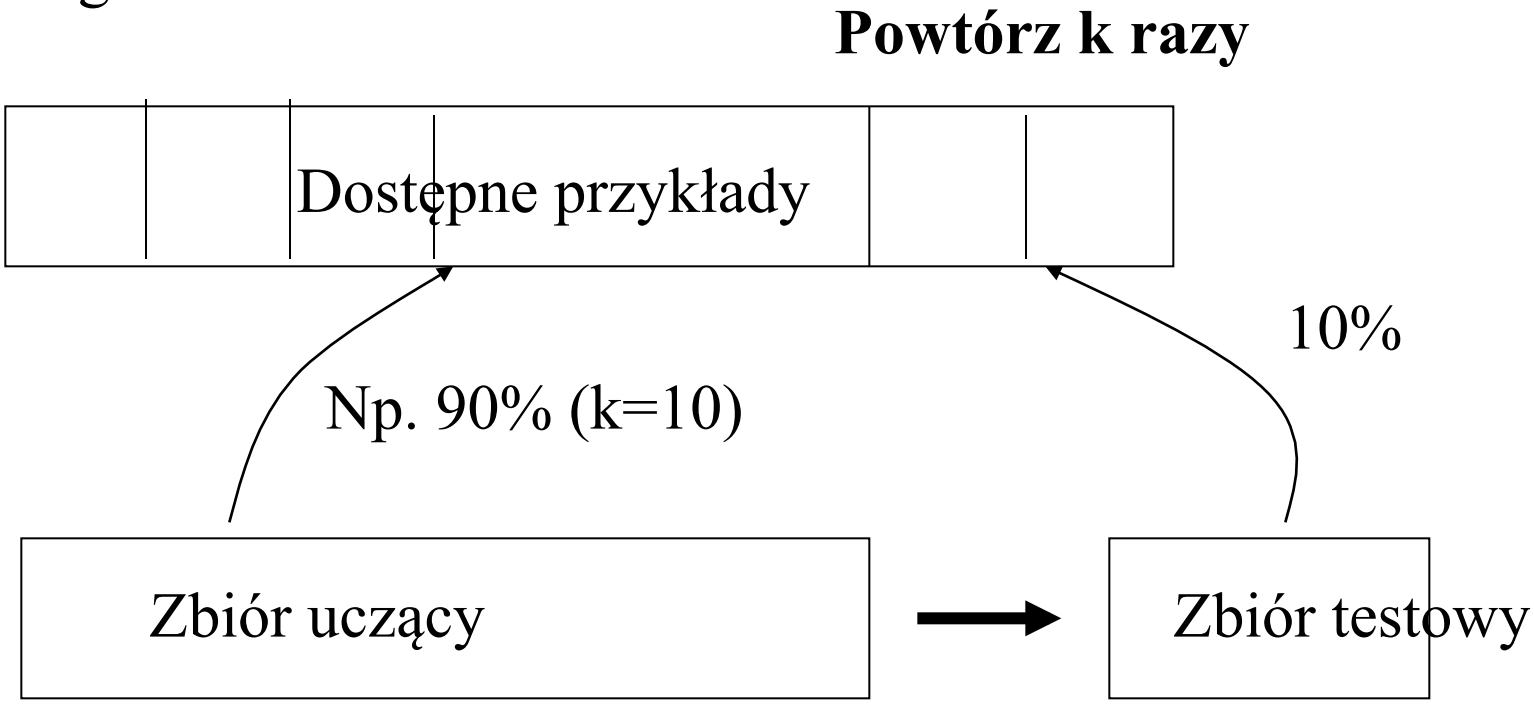
Wielokrotne podziały losowe



Po wielokrotnych podziałach losowych – oblicz wynik średni wybranej miary oceny każdego z klasyfikatorów

Mniejsza liczba przykładów (od 100 do kilku tysięcy)

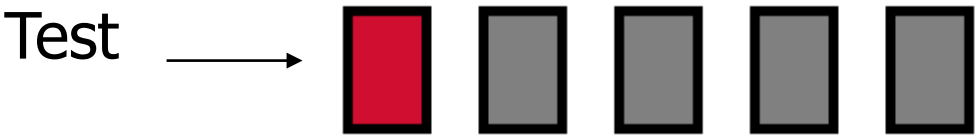
ang. k fold cross-validation



K –fold cross-validation

- Podziel losowo w k części (folds) w przybliżeniu tej samej wielkości

- Użyj jedenego podziału do testowania a reszty do budowy klasyfikatora



- Repeat k times

Uwagi o 10 fold cross-validation

- Stosuj wersję: stratified ten-fold cross-validation
- Dlaczego 10? Doświadczenie badaczy głównie eksperymentalne (zwłaszcza związane CART)
- Stratification – warstwowość ogranicza wariancje estymaty błędy!
- Lepsza wersja: repeated stratified cross-validation”
  - np. 10-fold cross-validation jest powtórzone kilka razy (z innym ziarnem rozkładu prawdopodobieństwa) i wynik średni z wielu powtórzeń.
  - Minimalizuje wariancje oszacowania

Losowanie warstwowe (stratified)

Pełen zbiór uczący etykietowany

np. 70% klasa 1 i 30% klasa 2

Podział losowy

Zbiór uczący

przybliż. 70% klasa 1 i 30% klasa 2

Zbiór testowy

przyb. 70% kl 1 i 30% kl 2

Podobne proporcje losowania klas w ew. zbiorze walidacyjnym

Zachowujemy proporcje klas w losowaniu

Przykład – C4.5 cross validation

C4.5 CRX (15 attributes, 490 training cases, 200 test cases)

Data Tree Rules Cross-validation Special Help

	Before pruning			After pruning			
Tree	Size	Errors	Errors (test)	Size	Errors	Errors (test)	Esti
1	101	18 (4.1%)	5 (10.2%)	50	28 (6.3%)	4 (8.2%)	15.
2	91	16 (3.6%)	9 (18.4%)	44	26 (5.9%)	9 (18.4%)	13.
3	95	16 (3.6%)	8 (16.3%)	48	23 (5.2%)	8 (16.3%)	13.
4	94	20 (4.5%)	8 (16.3%)	46	27 (6.1%)	7 (14.3%)	14.
5	102	17 (3.9%)	6 (12.2%)	51	26 (5.9%)	6 (12.2%)	14.
6	98	23 (5.2%)	11 (22.4%)	9	54 (12.2%)	5 (10.2%)	15.
7	112	21 (4.8%)	4 (8.2%)	41	30 (6.8%)	5 (10.2%)	14.
8	107	19 (4.3%)	13 (26.5%)	3	58 (13.2%)	8 (16.3%)	15.
9	88	25 (5.7%)	7 (14.3%)	40	29 (6.6%)	7 (14.3%)	14.
10	121	24 (5.4%)	7 (14.3%)	46	30 (6.8%)	7 (14.3%)	14.
Avg.	100.9	19.9 (4.5%)	7.8 (15.9%)	37.8	33.1 (7.5%)	6.6 (13.5%)	14.

Cross-validation (rules)

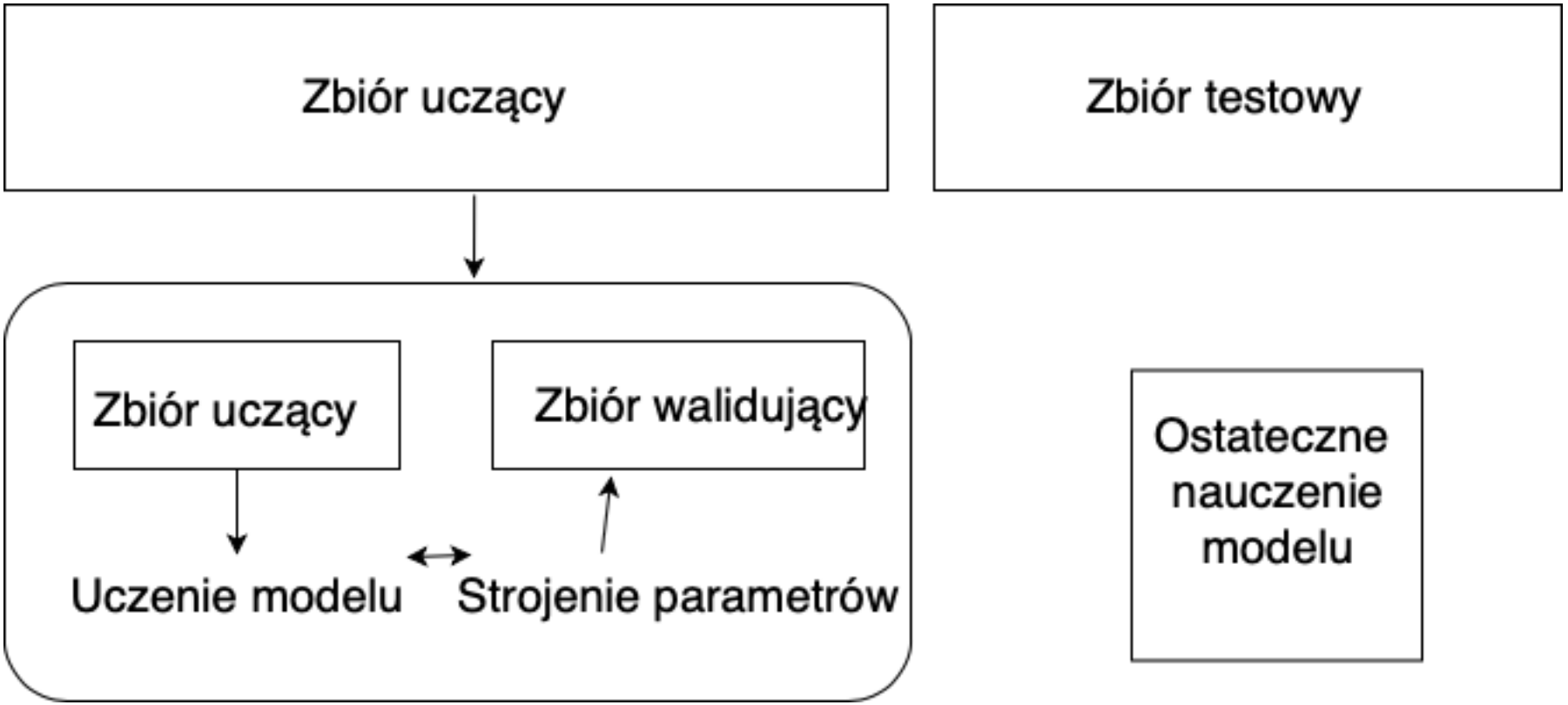
Ruleset	Size	Errors	Errors (test)
1	5	60 (13.6%)	1 (2.0%)
2	15	32 (7.3%)	10 (20.4%)
3	10	38 (8.6%)	9 (18.4%)
4	7	42 (9.5%)	7 (14.3%)
5	6	47 (10.7%)	5 (10.2%)
6	4	51 (11.6%)	6 (12.2%)
7	8	43 (9.8%)	6 (12.2%)
8	2	58 (13.2%)	8 (16.3%)
9	10	40 (9.1%)	6 (12.2%)
10	5	49 (11.1%)	7 (14.3%)
Avg.	7.2	46.0 (10.4%)	6.5 (13.2%)

Źródłó − aplikacja wenw. PP

Strojenie parametrów klasyfikatora i późniejsza ocena

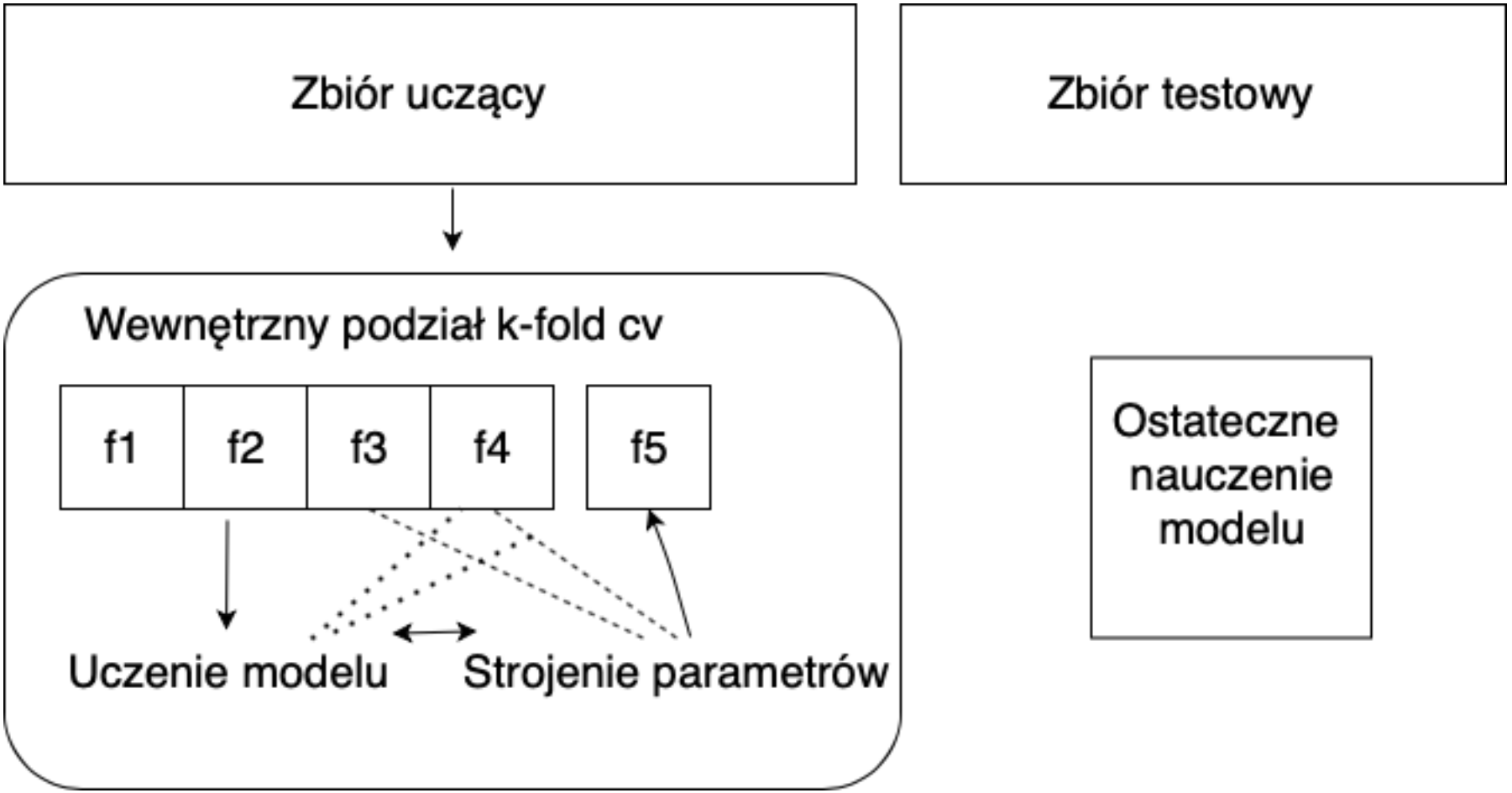
- Potrzeba specjalnego zbioru walidacyjnego, na którym prowadzi się eksperymentalne sprawdzanie wartości parametru (czasami wspomagany oprogramowaniem np. grid search)
  - Patrz np. redukcja (pruning) drzew, dobór k w algorytmie K-NN, strojenie parametrów ANN
- Wydzielony ze zbioru uczącego:
  - Właściwy zbiór uczący i walidujący = Niezależne od przykładów testowych
  - Moze być tzw. wewnętrzna (w zbiorze uczącym) ocena krzyżowa = wtedy podwójna pętla oceny (cross validations)

Strojenie klasyfikatora – potrzeba zbioru walidującego



Wydzielenie zbioru walidującego z części uczącej do doboru parametrów; Dla nich nauczenie klasyfikatora na pełnym zbiorze uczącycm

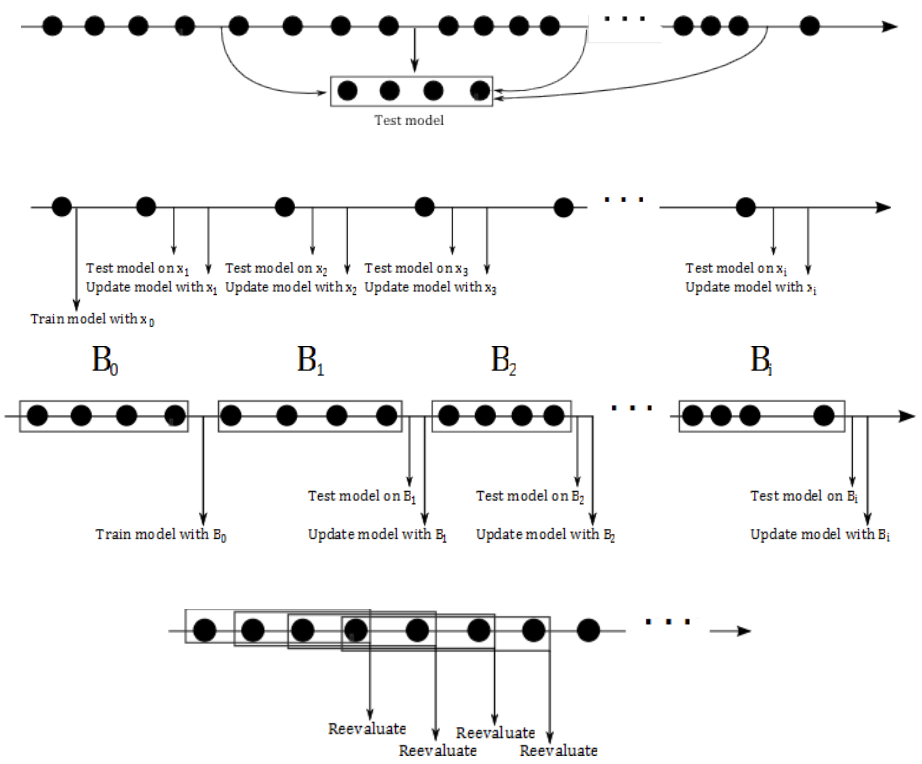
Strojenie klasyfikatora – wewnętrzne wielokrotne podziały



Wykorzystaj wewnętrzną k ocenę krzyżową − wymiana f bloków

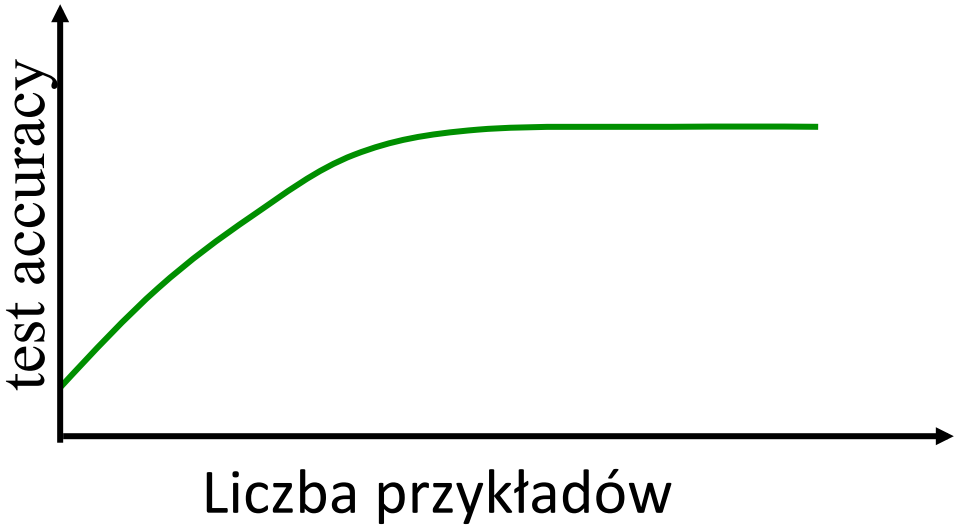
Ocena klasyfikatorów przyrostowych strumieniowych

- Holdout
  [np., Kirkby 2007]
- Test-then-train
  [np., Kirkby 2007]
- Block-based evaluation
  [np., Brzezinski & Stefanowski 2010]
- Prequential accuracy
  [Gama et al. 2013]
- Inne miary
  [Bifet & Frank 2010, Zliobaite et al. 2014]



Rysunek artykuł Brzeziński, Stefanowski Reacting to different types of concept drift: The accuracy updated ensemble algorithm

Krzywe przyrostowego uczenia się



Klasyfikatory przyrostowe – wizualizacja graficzna uczenia się w odniesieniu do kolejno dostępnych przykładów; Jeśli przyrostowo dostępne dane uczące są stacjonarne, dobre algorytmy powinny prowadzić do stopniowego przyrostu zdolności predykcyjnej

Inna sytuacja z tzw. zmiennych strumieniach danych − dryft definicji pojęcia (ang. concept dryft)

Porównywanie wielu klasyfikatorów

- Często należy porównać dwa klasyfikatory
- Uwaga: porównanie z niezależnością od danych?
  - Generatory losowe
  - Rzeczywiste dane (problem dependent)
- Oszacuj 10-fold CV estimates.
- Trudność: wariancja oszacowania.
- Możesz oczywiście zastosować „repeated CV”.
- Lecz, jak wiarygodnie ustalić konkluzję – który jest lepszy?

Porównywanie klasyfikatorów

- Jak oceniać skuteczność klasyfikacyjną dwóch różnych klasyfikatorów na tych samych danych?
- Ograniczamy zainteresowanie wyłącznie do trafności klasyfikacyjnej – oszacowanie techniką 10-krotnej oceny krzyżowej (ang. *k-fold cross validation*).
- Zastosowano dwa różne algorytmy uczące *AL*1 i *AL*2 do tego samego zbioru przykładów, otrzymując dwa różne klasyfikatory *KL*1 i *KL*2. Oszacowanie ich trafności klasyfikacyjnej (10-fcv):
  - klasyfikator *KL*1 $\rightarrow$ 86,98%
  - klasyfikator *KL*2 $\rightarrow$ 87,43%.
- Czy uzasadnione jest stwierdzenie, że klasyfikator *KL*2 jest skuteczniejszy niż klasyfikator *KL*1?

Analiza wyniku oszacowania trafności klasyfikowania

Podział	Kl_1	Kl_2
1	87,45	88,4
2	86,5	88,1
3	86,4	87,2
4	86,8	86
5	87,8	87,6
6	86,6	86,4
7	87,3	87
8	87,2	87,4
9	88	89
10	85,8	87,2
Srednia	**86,98**	**87,43**
Odchylenie	**0,65**	**0,85**

- Test statystyczny (t-Studenta dla par zmiennych/zależnych)
- H0 : średnie oceny kl1 i kl2 się nie różnią znacząco
- H1: średnia ocena jednego z klasyfikatorów jest wyższa niż drugiego
- *t*emp = 1,733 (p = 0,117) ???
- ALE !!! W art. naukowych zastosuj odpowiednie poprawki

przy wykonaniu testu (kwestia naruszenia założeń co do rozkładu *t*).

Porównanie działania dwóch klasyfikatorów DT oraz n2 na wielu zbiorach danych (wyniki średnie z 10-oceny krzyżowej wraz z przedziałem ufności α=0,95)

Data set	Classification accuracy *DT* (%)	Classification accuracy n^2 (%)	Improvement n^2 vs. *DT* (%)
Automobile	**85.5 ± 1.9**	**87.0 ± 1.9**	1.5*
Cooc	**54.0 ± 2.0**	**59.0 ± 1.7**	5.0
Ecoli	**79.7 ± 0.8**	**81.0 ± 1.7**	1.3
Glass	**70.7 ± 2.1**	**74.0 ± 1.1**	3.3
Hist	**71.3 ± 2.3**	**73.0 ± 1.8**	1.7
Meta-data	**47.2 ± 1.4**	**49.8 ± 1.4**	2.6
Primary Tumor	**40.2 ± 1.5**	**45.1 ± 1.2**	4.9
Soybean-large	**91.9 ± 0.7**	**92.4 ± 0.5**	0.5*
Vowel	**81.1 ± 1.1**	**83.7 ± 0.5**	2.6
Yeast	**49.1 ± 2.1**	**52.8 ± 1.8**	3.7

Źródło – własny artykuł naukowy

Dalsze porównania klasyfikatorów

- Dwa modele na wielu zbiorach danych – test rangowy Wilcoxona
  - Detale za chwile
- Wiele modeli/klasyfikatorów na wielu zbiorach danych
  - Test Friedmana (odpowiada na H0: że nie ma znaczących różnic w ocenie klasyfikatorów; H1 negacja);
  - Jeśli odrzucimy H0, przedstaw średnie rangi przypisane każdemu klasyfikatorowi;
  - Wykonanie posthoc analizy (np. Nemenyi) − policzenie CD krytycznej różnicy rang

Globalna ocena (2 alg. wiele zb. danych)

Wilcoxon test (sparowany test rangowy)

H0: nie ma różnicy oceny klasyfikatorów

1. Różnice oceny klasyfikatorów uporządkuj wg. wartości bezwzględnych i przypisz im rangi.
2. R+ suma rang dla sytuacji gdy klasyfikator 1 jest lepszy niż klasyfikator 2 // R- sytuacja odwrotna
3. Oblicz statystykę T = min{R+;R-}

   Rozkład T jest stabelaryzowany / prosta reguła decyzyjna
4. Dla odpowiednio dużej liczby *m* zbiorów danych można stosować przybliżenie z

$$z = \frac{\min\{R+;R-\} - \frac{1}{4}m(m-1)}{\sqrt{\frac{1}{24}m(m+1)(2m+1)}}$$

Porównanie dwóch klasyfikatorów

Dane	Klasyf B	Klas M	Różnica	ranga
D1	0,763	0,768	+0,005	3,5
D2	0,599	0,591	-0.008	7
D3	0,954	0,971	+0,017	9
…	…	…	…	…
D12	0,619	0,666	+0,047	13
D13	0,972	0,981	+0,009	8
D14	0,957	0,978	+0,021	10

- Obliczenie średnich rang R+=3,5+9+12+5+6+14+11+13+8+10+1,5 = 83
- R- = 7 + 3,5 +1,5 = 12
- Z = -2.51 < - 1,96 / H0 odrzucamy, klasyfikator M średnio lepszy niz Klasyfikator B

Test Friedmana

- H0: oceny wszystkich klasyfikatorów nie różnią się
- H1: Oceny niektórych klasyfikatorów są lepsze niż pozostałych

Dla każdego zbioru danych (i=1,..,N) ustawiamy rangi m klasyfikatorów wg. ich rezultatów

Następnie oblicz średnie rangi klasyfikatorów r_j (j=1,..m)

Statystyka Friedmana ma rozkład χ^2 z N-1 stopniami swobody

$$\chi_F^2 = \frac{12m}{N(N+1)}\left(\sum_{j=1}^{m} r_j^2 - \frac{N(N+1)^2}{4}\right)$$

Jeśli odrzucamy H0, to liczymy post-hoc analize (np. Nemeyi test)

$$CD = q_\alpha \sqrt{\frac{N(N+1)}{6m}}$$

Algorytmy z różnica średnich rang większą niż CD są statystycznie lepsze

Test Friedmana

Dane	Klasyfikator1	Klasyfikator2	Klasyfiator3
Zb danych 1	1	3	2
Zb danych 2	1,5	1,5	3
Zb danych 3	1	2	3
Zb danych 4	2	3	1
Zb danych 5	2,5	2,5	1
Średnie rangi	1,6	2,4	2,0

Fobl = 37,1 a krytyczna statystkak 9,488 = odrzucamy H0

CD wartość krytyczna - klasyfikaory sa nierozróżnialne

Podejścia teoretyczne

- Obliczeniowa teoria uczenia się (COLT)
  - **PAC** model (Valiant)
  - Wymiar Vapnik Chervonenkis → VC Dimension
- Pytania o ogólne prawa dotyczące procesu uczenia się klas pewnych funkcji z przykładów - rozkładów prawdopodobieństwa.
- Silne założenia i ograniczone odniesienia do problemów praktycznych.

Perspektywa opisowa

- Trudniejsza niż ocena zdolności klasyfikacyjnych.
- Rozważmy przykład reguł:
  - Klasyfikacyjne

  Jeżeli (atr1=wartość) and (atr3=wartość) to (klasa=A)

  - Asocjacyjne.

    Jeżeli ACD to B
- Pojedyncza reguła oceniana jako potencjalny reprezentant „interesującego” wzorca z danych
  - W literaturze propozycje tzw. ilościowych miar oceny reguł oraz sposoby definiowania „interesujących” reguł, także na podstawie wymagań podawanych przez użytkownika.

Przykład reguł klasyfikacyjnych

Minimalny zbiór pewnych reguł

- *if* (*a*2 = s) ∧ (*a*3 ≤ 2) *then* (*d* = C1) {*x*1,*x*7}
- *if* (*a*2 = n) ∧ (*a*4 = c) *then* (*d* = C1) {*x*3,*x*4}
- *if* (*a*2 = w) *then* (*d* = C2) {*x*2,*x*6}
- *if* (*a*1 = f) ∧ (*a*4 = a) *then* (*d* = C2) {*x*5,*x*8}

Reguła z conf < 1

- *if* (*a1*=m) *then* (*d*=C1) {*x1,x3,x7 | x6*} 3/4

id.	a_1	a_2	a_3	a_4	d
x_1	m	s	1	a	C1
x_2	f	w	1	b	C2
x_3	m	n	3	c	C1
x_4	f	n	2	c	C1
x_5	f	n	2	a	C2
x_6	m	w	2	c	C2
x_7	m	s	2	b	C1
x_8	f	s	3	a	C2

Opisowe miary oceny reguł

- Miary dla reguły *r* (*jeżeli P to Q*) definiowane na podstawie zbioru przykładów *U*, z którego została wygenerowana.
- Tablica kontyngencji dla reguły *jeżeli P to Q* :

	Q	$\neg Q$	
P	n_{PQ}	$n_{\mathrm{P}\neg\mathrm{Q}}$	n_{P}
$\neg P$	$n_{\neg\mathrm{PQ}}$	$n_{\neg\mathrm{P}\neg\mathrm{Q}}$	$n_{\neg\mathrm{P}}$
	n_{Q}	$n_{\neg\mathrm{Q}}$	n

- Przegląd różnych miar, np.: Ya Y.Y, Zhong N.: An analysis of quantitative measures associated with rules, w: Proc. of the Third Pacific-Asia Conference on Knowledge Discovery and Data Mining, LNAI 1574, Springer, 1999, s. 479-488.
- Także rozprawa habilitacyjna J.Stefanowski: Algorytmy indukcji reguł w odkrywaniu wiedzy (dostępna przez WWW) oraz rozprawa doktorska p. Izabeli Szczęch.

Popularne miary oceny reguł

- Wsparcie reguły jeśli P to Q (ang. *support*) zdefiniowane jako:

$$G(P \wedge Q) = \frac{n_{PQ}}{n}$$

- Dokładność (ang. *rule accuracy*) / wiarygodność (ang. *confidence*) reguły (bezwzględne wsparcie konkluzji *Q* przez przesłankę *P*):

$$AS(Q \mid P) = \frac{n_{PQ}}{n_P}$$

- Względne pokrycie (ang. *coverage*) reguły zdefiniowane jako:

$$AS(P \mid Q) = \frac{n_{PQ}}{n_Q}$$

Zaawansowane miary oceny reguł

Change of support – rodzaj konfirmacji wsparcia hipotezy Q przez wystąpienie przesłanki P (propozyjca Piatetsky-Shapiro)

$$CS(Q \mid P) = AS(Q \mid P) - G(Q)$$

gdzie $$G(Q) = \frac{n_Q}{n}$$

Zakres wartości od -1 do +1 ; Interpretacja: różnica między prawdopodobieństwami a prior i a posterior; dodatnie wartości wystąpienie przesłanki P powoduje konkluzję Q; ujemna wartość wskazuje że nie ma wpływu.

Degree of independence:

$$IND(Q,P) = \frac{G(P \wedge Q)}{G(P) \cdot G(Q)}$$

Złożone miary oceny reguł

Połączenie miar podstawowych

Significance of a rule (propozycja Yao i Liu)

$$S(Q \mid P) = AS(Q \mid P) \cdot IND(Q, P)$$

Klosgen's measure of interest

$$K(Q \mid P) = G(P)^{\alpha} \cdot (AS(Q \mid P) - G(Q))$$

Michalski's weighted sum

$$WSC(Q \mid P) = w_1 \cdot AS(Q \mid P) + w_2 \cdot AS(P \mid Q)$$

The relative risk (Ali, Srikant):

$$r(Q \mid P) = \frac{AS(Q \mid P)}{AS(Q \mid \neg P)}$$

Przykład diagnostyki technicznej

- Bada się stan techniczny 76 autobusów tego samego typu (dokładnie ich silników) na podstawie symptomów stanu technicznego - parametrów pochodzących z okresowych badań diagnostycznych [dane prof. J.Zak, analiza J.Stefanowski]
  - Autobusy są podzielone na dwie klasy: dobry i zły stan techniczny pojazdu
- Cel analizy
  - Ocenia się jakość diagnostyczną symptomów stanu technicznego
  - Poszukuje się zależności pomiędzy wartościami najistotniejszych w tych symptomów a przydziałem do klas = konieczność interpetacji wzorców w postaci reguł
  - Konstruuje się klasyfikator stanu technicznego

Rozważane symptomy

s1 – prędkość maksymalna [km/h],

s2 – ciśnienie sprężania [Mpa],

s3 – zawartość elementów smołowatych w spalinach wylotowych [%],

s4 – moment obrotowy silnika [Nm],

s5 – letnie zużycie paliwa [l/100lm],

s6 – zimowe zużycie paliwa [l/100km],

s7 – zużycie oleju [l/1000km],

s8 – aktualna moc silnika [KM].

Dwie klasy decyzyjne:

1. Autobusy z silnikami w dobrym stanie – dalsza eksploatacja (46),
2. Autobusy z silnikami w złym stanie – konieczność napraw (30).

Minimalny zbiór reguł klasyfikujących

1. if (*s2*≥2.4 MPa) & (*s7*<2.1 /*/*1000km) then (technical state=good) [46]

2. if (*s2*<2.4 MPa) then (technical state=bad) [29]

3. if (*s7*≥2.1 /*/*1000km) then (technical state=bad) [24]

Oszacowana trafność klasyfikowania **(‘leaving one out’ test)** 98.7%**.**

Lecz trudność ich interpretacji

Poszukiwanie innych reguł z danych

Próg satysfakcji dla miary support (51%):

1. if (*s1*>85 km/h) then (technical state=good) [34]
2. if (*s8*>134 kM) then (technical state=good) [26]
3. if (*s2*≥2.4 MPa) & (*s3*<61 %) then (technical state=good) [44]
4. if (*s2*≥2.4 MPa) & (*s4*>444 Nm) then (technical state=good) [44]
5. if (*s2*≥2.4 MPa) & (s7<2.1 *l*/1000km) then (technical state=good) [46]
6. if (*s3*<61 %) & (*s4*>444 Nm) then (technical state=good) [42]
7. if (*s1*≤77 km/h) then (technical state=bad) [25]
8. if (*s2*<2.4 MPa) then (technical state=bad) [29]
9. if (*s7*≥2.1 *l*/1000km) then (technical state=bad) [24]
10. if (*s3*≥61 %) & (*s4*≤444 Nm) then (technical state=bad) [28]
11. if (*s3*³61 %) & (*s8*<120 kM) then (technical state=bad) [27]

Uwagi do źródeł

Wykorzystano książki:

- S.Weiss, C.Kulikowski: Computer Systems That Learn: Classification and Prediction Methods from Statistics, Neural Nets, Machine Learning and Expert Systems, Morgan Kaufmann 1991.
- **N.Japkowicz, M. Shah: Evaluating Learning Algorithms: A Classification Perspective, Cambridge Presss 2011**.
- I.Konennko, M.Kukar: Machine Learning and Data Mining, 2007.
- J.Han, M.Kember: Data mining. Morgan Kaufmann 2001.

oraz inspiracje ze slajdów wykładów:

- J.Han; G.Piatetsky-Shapiro; D.Page, A.Avati + materiały związane z WEKA i prezentacji W.Kotłowski nt. Statistical Analysis of Computational Experiments in Machine Learning

Wybrane artykuły

- Patrz następny slajd

Wybrane artykuły

1. Kohavi, R. (1995): A study of cross-validation and bootstrap for accuracy estimation and model selection. *Proc. of the 14th Int. Joint Conference on Artificial Intelligence*, 1137—1143.
2. Salzberg, S. L. (1997): On comparing classifiers: Pitfalls to avoid and a recommended approach. *Data Mining and Knowledge Discovery*, 1, 317—328.
3. Dietterich, T. (1998): Approximate statistical tests for comparing supervised classification learning algorithms. *Neural Computation*, 10:7, 1895-–1924.
4. Bouckaert, R. R. (2003): Choosing between two learning algorithms based on calibrated tests. *ICML 2003*.
5. Bengio, Y., Grandvalet, Y. (2004): No unbiased estimator of the variance of k-fold cross-validation. *Journal of Machine Learning Research*, 5, 1089—1105.
6. Demsar, J. (2006): Statistical comparisons of classifiers over multiple data sets. *Journal of Machine Learning Research*, 7, 1–30.
7. S. Raschka (2018) Model Evaluation, Model Selection, and Algorithm Selection in Machine Learning, arXiv 2018
8. Sesja spacjalna nt. oceny systemów uczących (N.Japkowicz) na ICML 2007 + tutorial

Oraz nowsze artykuły, np. o testach statystycznych Salvador Garcia Univ. Granada

Pytanie i komentarze?

Dalszy kontakt:

jerzy.stefanowski@cs.put.poznan.pl

http://www.cs.put.poznan.pl/jstefanowski/

Fundusze Europejskie Polska Cyfrowa

Rzeczpospolita Polska

Unia Europejska Europejski Fundusz Rozwoju Regionalnego