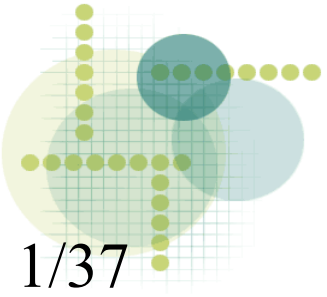
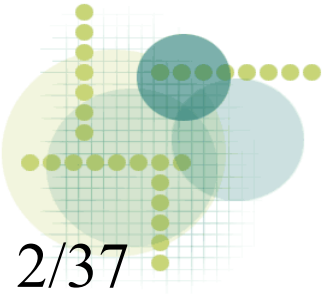
Kombinacja jądrowych estymatorów gęstości w klasyfikacji - przegląd literatury

Mateusz Kobos, 22.10.2008
Seminarium Metody Inteligencji Obliczeniowej



Spis treści



Klasyfikator Bayesowski

Klasyfikator Bayesowski:

$$P(c = i|x) = \frac{p(x|c = i)P(c = i)}{p(x)}$$

gdzie:

- $P(c = i|x)$ - prawdopodobieństwo, że punkt x należy do klasy i
- $p(x|c = i)$ - gęstość dla rozkładu punktów z klasy i w punkcie x
- $P(c = i)$ - prawdopodobieństwo pojawienia się punktu z klasy i
- $p(x) = \sum_{j=1}^{K} p(x|c = j)P(c = j)$ - gęstość w punkcie x
  - K - liczba klas

- Teoretycznie, by dokonać klasyfikacji wystarczy „tylko” znać gęstość punktów każdej z klas

Estymator jądrowy (Kernel Density Estimator (KDE))

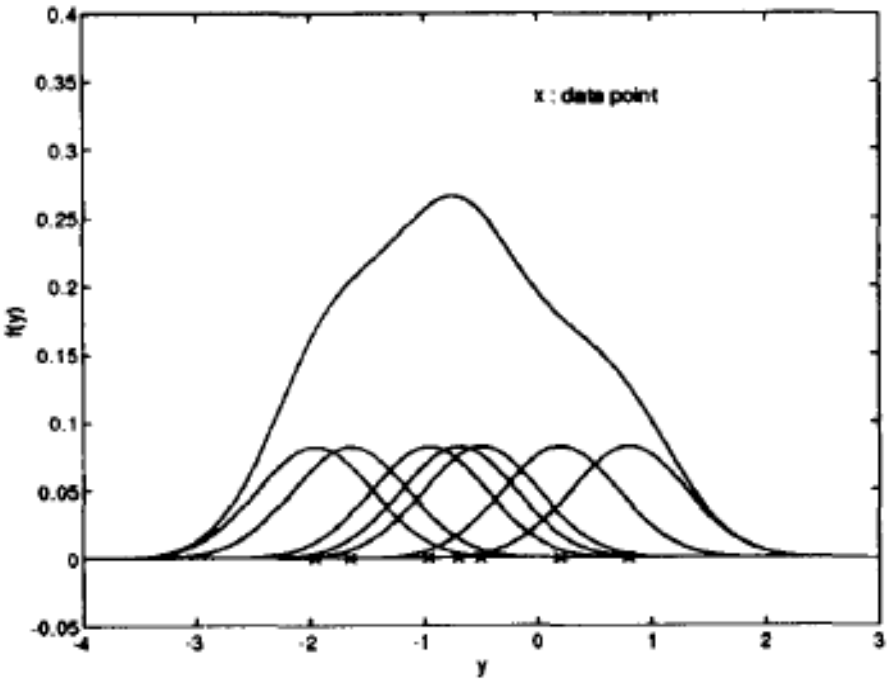
Estymator jądrowy:

$$p(x) = \frac{1}{n}\sum_{i=1}^{n}\frac{1}{h^d}\phi(\frac{x - x_i}{h})$$

gdzie:

- x - punkt, w którym dokonujemy estymacji
- x_i - punkty należące do zbioru uczącego
- n - liczba punktów
- h - "szerokość jądra"/"skala"/"bandwidth"
- d - liczba wymiarów

Pobrane z [Hwang94]



Gaussian Mixture Model (GMM)

Gaussian Mixture Model:

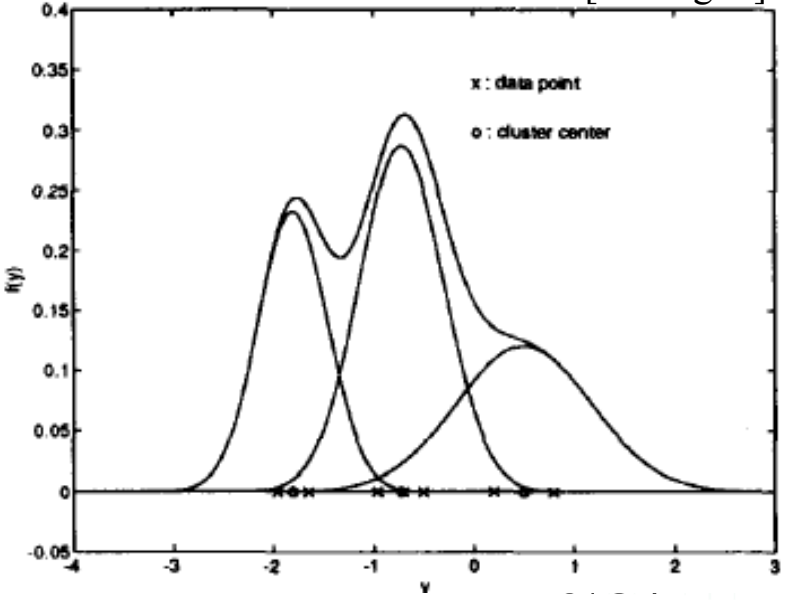
$$p(x) = \sum_{i=1}^{k} P(C = i)p(x|C = i)$$

gdzie:

- $P(C = i)$ - pr. wylosowania i-tego rozkładu
- $p(x|C = i) = N(\mu_i, \Sigma_i)$ - pr. wylosowania punktu x z i-tego rozkładu.

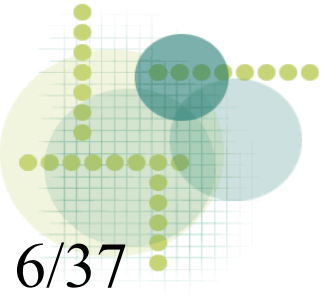
- Parametry (μ, Σ_i) dobiera się najczęś
  za pomocą algorytmu EM korzystające
  z metody największej wiarogodności
- Liczba składowych jest z góry zadana

Pobrane z [Hwang94]



Kombinacja KDE (KDEC)

- Ogólny pomysł: Estymować gęstość poprzez kombinację (liniową) estymacji za pomocą KDE o różnej szerokości jądra



Kombinacja KDE (KDEC)

- Wstępny pomysł (algorytm KDEC):
  - Dobierać szerokości jąder dla kolejnych KDE tak, by malały wykładniczo w zależności od parametru a. Szerokość jądra określona jako:

$$h_k = a^k h_0$$

gdzie:

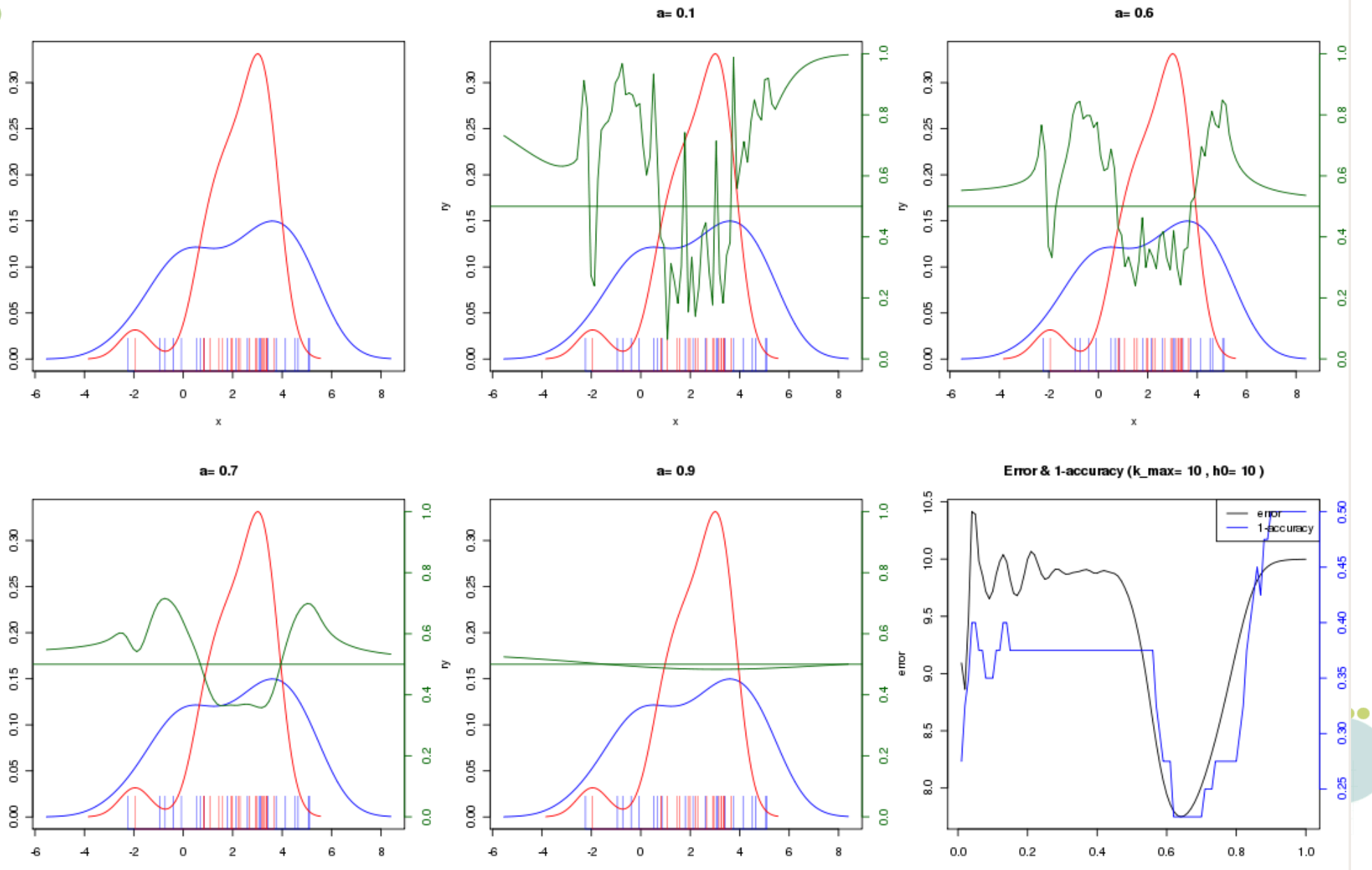
- h - szerokość jądra dla k-tego KDE
- a - parametr
- h_0 - stała ew. parametr

  - Każda z KDE ma taką samą wagę
  - Minimalizujemy funkcję błędu (za pomocą cross-validation lub np. metodą największego spadku z momentum)

Kombinacja KDE (KDEC)

- Parametry modelu:
  - a – jak szybko maleją jądra
  - h_0 – największa szerokość jądra (tu: ustalone)
  - w_i – waga każdego z KDE w kombinacji liniowej (tu: ustalone i takie same)
- Potencjalne modyfikacje:
  - Dobierać kształt jąder dla każdej klasy oddzielnie
  - Określić inną funkcję zmniejszania się jąder
  - Optymalizować względem innych parametrów
  - Ustalać h_0 na podstawie któregoś z popularnych kryteriów doboru szerokości jądra

Kombinacja KDE (KDEC) – przykładowy wykres błędu

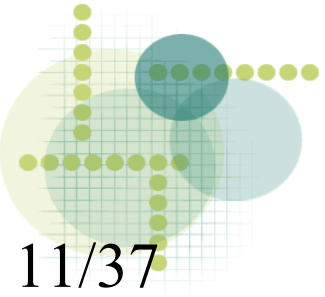


Opinie o klasyfikacji poprzez estymację gęstości

- Dla danych o dużej liczbie wymiarów estymacja gęstości jest b. trudna, bo dla skończonego zbioru danych w miejscach o dużej gęstości może być b. mało punktów [Bishop95]
- W klasyfikacji oddzielne uczenie się rozkładu każdej z klas może być niepotrzebne i mylące. Skupiamy się na dokładniejszym dostosowywaniu rozkładu do pewnych punktów, jednak te punkty mogą nie mieć znaczenia dla obliczania prawdopodobieństwa przynależności do klasy. Dla klasyfikacji, prawdopodobieństwo musimy dobrze estymować tylko w pobliżu granic decyzyjnych.[Hastie01]

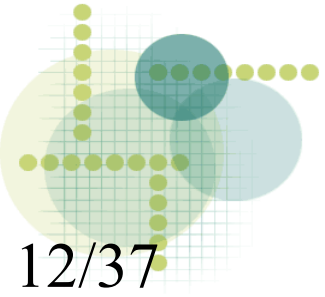
Opinie o klasyfikacji poprzez estymację gęstości

- Estymacja gęstości jest centralnym problemem modelowania danych i uczenia maszynowego. Jest to problem trudniejszy od klasyfikacji i regresji [Shawe-Taylor07]
- Chociaż optymalne estymatory gęstości nie koniecznie prowadzą do dobrej klasyfikacji, to dobre estymacje klas z pewnością dają klasyfikatory o niskim błędzie [Hoti04]



Opinie o klasyfikacji poprzez estymację gęstości

- Dobra estymacja gęstości za pomocą estymatorów jądrowych jest możliwa do 6 wymiarów. Jednak, mimo że sama estymacja gęstości może być zła, to klasyfikacja na jej podstawie może być dobra. Metody jądrowe w zagadnieniu klasyfikacji dają dobre wyniki również dla dziesiątków wymiarów [Scott04]



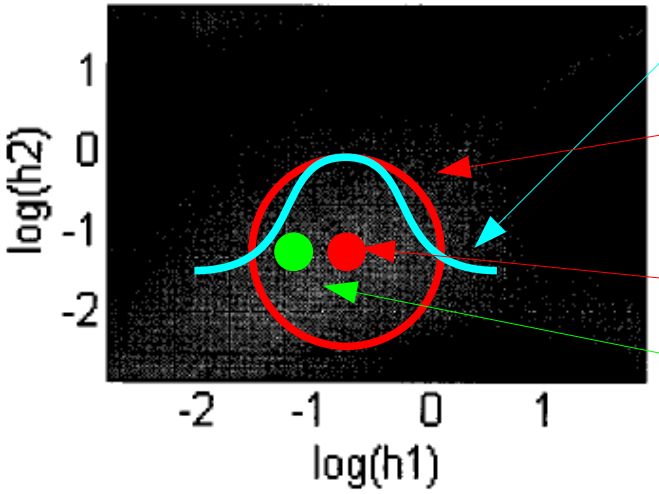
Artykuł: klasyfikacja za pomocą kombinacji KDE

- Artykuł [Ghosh06a]: Anil K. Ghosh, Probal Chaudhuri, and Debasis Sengupta, „Classification Using Kernel Density Estimates: Multi-scale Analysis and Visualization”, Technometrics, 2006
- Pomysł: klasyfikacja za pomocą kombinacji KDE o różnej szerokości jądra
  - Modyfikacja tego pomysłu dla estymatorów gęstości opartych na metodzie k-NN: [Ghosh06b]

Artykuł: klasyfikacja za pomocą kombinacji KDE cd.

- Cechy metody:
  - Zagadnienie klasyfikacji binarnej
  - Każdej z klas przypisujemy oddzielną szerokość jądra (**h1** dla klasy 1, **h2** dla klasy 2)
- Działanie algorytmu - ogólnie:
  1. Normalizujemy punkty dla każdej z klas oddzielnie
  2. Bierzemy najlepsze pary szerokości jąder **h1**, **h2**: tzn. te, które dają globalnie najmniejszy błąd klasyfikacji. Używamy pomocą cross-validation.
  3. Dla danego punktu testowego **x** obliczamy **P(c=1|x)** jako ważoną sumę **P(c=1|x)** dla najlepszych par **h1**, **h2**

wykres: 1-błąd klasyfikacji



Wartości wag

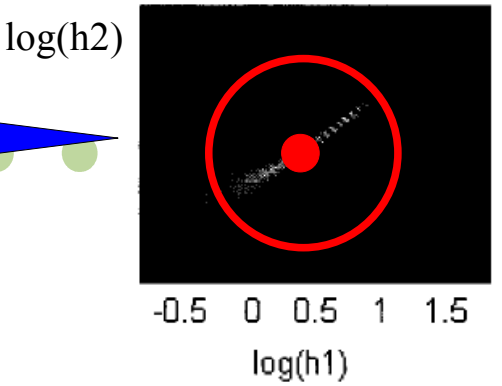
Zakres niezerowych wartości wag

Optymalny punkt klasyfikacji

Optymalny punkt estymacji gęstości

Unormowane mnożenie

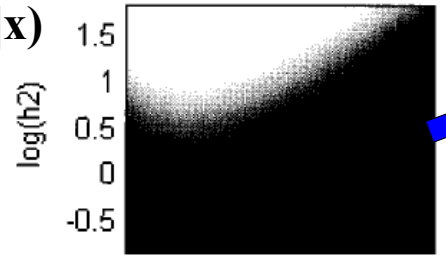
wykres: „adjusted weighted p-value”



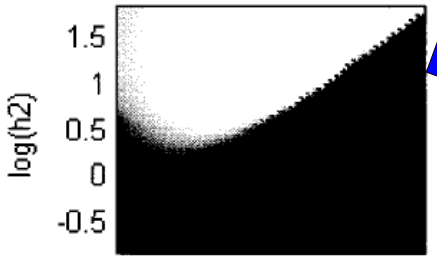
mnożenie

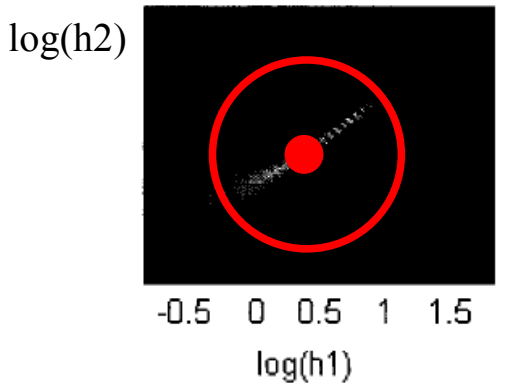
Wartości dla punktu testowego **x**

wykres: **P(c=1|x)**



wykres: p-value przypisania do klasy **c=1**





sumuj i normuj

wynik: **P(c=1|x)** = 0.7

Obrazki pobrane z [Ghosh06a]

Artykuł: klasyfikacja za pomocą kombinacji KDE - porównanie

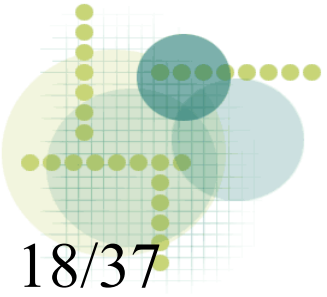
- Artykuł: brany jest punkt optymalny pod względem błędu cross-validation w przestrzeni szerokości jąder i jego najbliżsi sąsiedzi
  - Wada: parametry modelu są dla sąsiadów podobne
    - lepsze wyniki można by prawdopodobnie uzyskać biorąc bardziej odległe punkty (większa różnorodność)
  - KDEC: branych jest kilka odległych od siebie punktów w jej przestrzeni (odległości między punktami są dobierane poprzez minimalizację funkcji błędu)

Artykuł: klasyfikacja za pomocą kombinacji KDE - porównanie

- Artykuł: wagi każdego z elementów kombinacji dobierane są za pomocą gęstości normalnej
  - KDEC: wagi są identyczne
- Artykuł: przeszukiwana jest przestrzeń (h1, h2)
  - KDEC: przeszukiwana jest przestrzeń (a) (ew. (a1, a2)

Artykuł: klasyfikacja za pomocą kombinacji KDE - porównanie

- Artykuł: Używana jest brutalna metoda optymalizacji – sprawdzanie wartości funkcji błędu w każdym z punktów (pewnej siatki)
  - KDEC: można minimalizować funkcję błędu tak jak w MLP: np. metodą największego spadku z momentum



Artykuł: estymacja gęstości za pomocą stacking z KDE i GMM

- Artykuł [Smyth99]: Padhraic Smyth, David Wolpert, „Linearly Combining Density Estimators via Stacking”, Machine Learning, 1999
- Pomysł: estymacja gęstości za pomocą kombinacji liniowej GMM i KDE. Parametry kombinacji obliczane za pomocą techniki meta-uczenia: stacking i alg. EM.

Artykuł: estymacja gęstości za pomocą stacking z KDE i GMM cd.

- Do zagadnienia estymacji za pomocą kombinacji estymatorów podchodzą od strony algorytmów meta-uczących się.
- Działanie algorytmu:
  - Standardowy algorytm stacking z cross-validation:
    - modelami poziomu 0 są estymacje gęstości (KDE, GMM)
    - modelem poziomu 1 (poziom łączący) jest kombinacja liniowa z parametrami dobieranymi za pomocą alg. EM tak, by maksymalizować log-likelihood punktów ze zbioru uczącego

Artykuł: estymacja gęstości za pomocą stacking z KDE i GMM cd.

- Działanie algorytmu dla KDE:
  1. Wykonaj estymację dla określonych predefiniowanych szerokości jąder KDE przy pomocy cross-validation.
  2. Oblicz parametry kombinacji liniowej estymatorów za pomocą EM.
  3. Wykonaj estymację każdym z estymatorów KDE tym razem na całym zb. uczącym.
  4. Połącz estymacje za pomocą obliczonej kombinacji liniowej i zwróć wynik.

Artykuł: estymacja gęstości za pomocą stacking z KDE i GMM - porównanie

- Artykuł: nie jest dokonywana klasyfikacja, optymalizowane jest log-likelihood dopasowania do danych
  - KDEC: dokonujemy klasyfikacji i minimalizujemy błąd związany z klasyfikacją
- Artykuł: szerokości jąder są predefiniowane (chociaż proponują, że można by też dobierać je dynamicznie)
  - KDEC: szerokości jąder dobierane dynamicznie

Artykuł: estymacja gęstości za pomocą stacking z KDE i GMM - porównanie

- Artykuł: każda z estymacji składowych (KDE i GMM) jest wykonywana za pomocą cross-validation, co przeciwdziała przeuczeniu
- Artykuł: stosowane są KDE i GMM
  - KDEC: stosowane są tylko KDE
- Artykuł: każda z estymacji składowych ma inną wagę
  - KDEC: wszystkie wagi takie same

Artykuł: Filtered KDE

- Artykuł [Marchette96]: David J. Marchette, Carey E. Priebe, George W. Rogers, Jeffrey L. Solka „Filtered Kernel Density Estimation”, Computational Statistics, 1996
- Pomysł: Połączenie wielu KDE z GMM. Na całej przestrzeni określamy wagi każdego z KDE.

Artykuł: Filtered KDE cd.

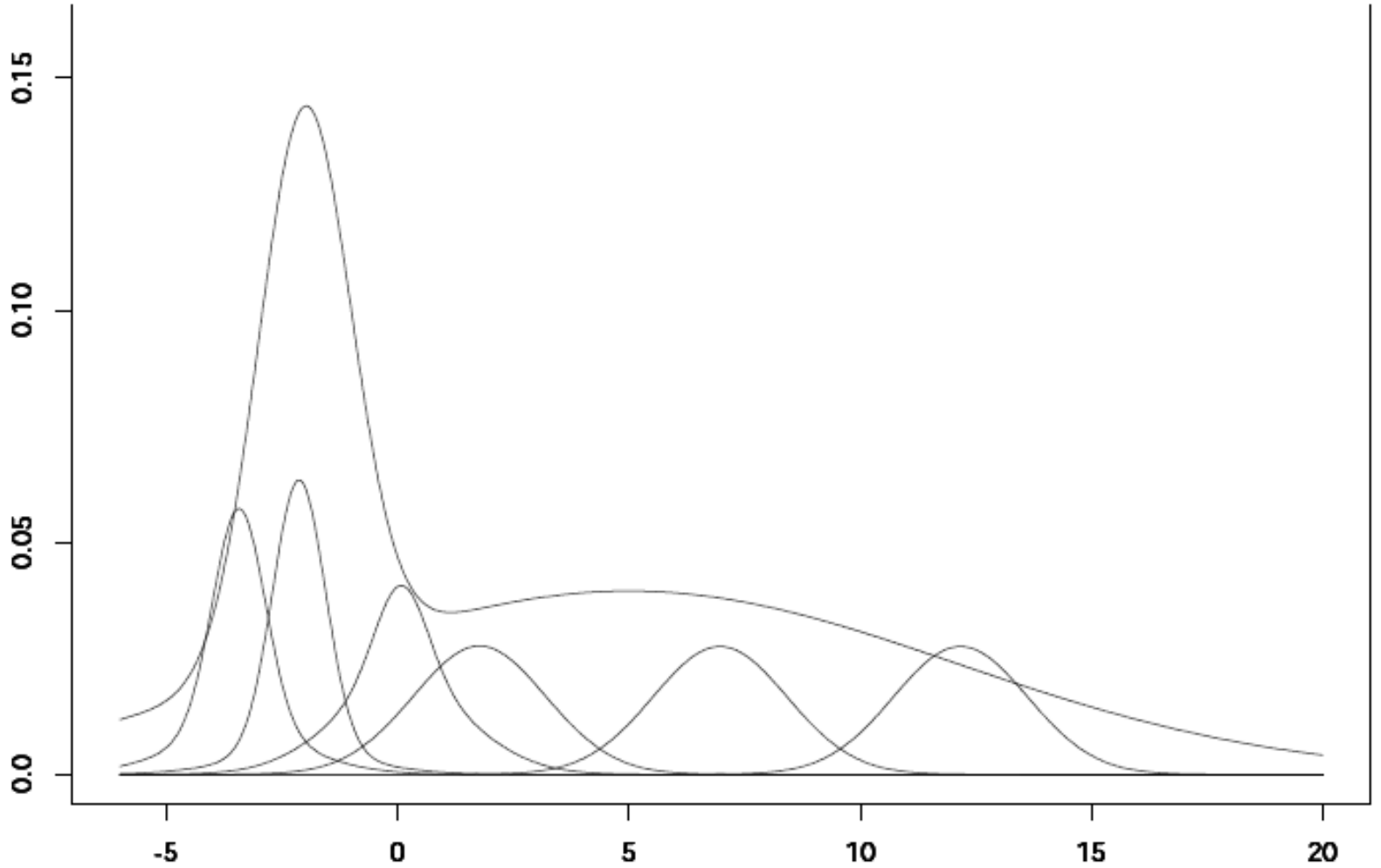
- Pomysł – bardziej szczegółowo:
  - Jest podanych kilka predefiniowanych szerokości jąder odpowiadających estymatorom KDE.
  - Do każdego KDE jest przypisany rozkład wagi, który mówi, w których miejscach dziedziny dana szerokość jest ważna, a w których nie. Dzięki temu możemy definiować obszary, w których jądra mają być szerokie i te, w których mają być wąskie.
  - Gdy obliczamy gęstość w jakimś punkcie, to obliczamy ważoną sumę KDE zgodnie z rozkładem wag w danym punkcie.

Artykuł: Filtered KDE cd.

- W praktyce rozkład wag określamy poprzez GMM – każda składowa odpowiada dominacji jednej z szerokości jądra w KDE.
- Kształty jąder w danym obszarze są określane na jeden z 2 sposobów:
  - przez optymalizację w ramach jednego obszaru
  - przez użycie wariancji punktów obszaru (w praktyce wyniki tych obu sposobów są podobne).

Artykuł: Filtered KDE cd.

- Przykład: 2 składowe GMM; szerokości jąder modelu w różnych punktach



Pobrane z [Marchette96]

Artykuł: Filtered KDE - porównanie

- Artykuł: nie jest optymalizowana jakość klasyfikacji
- Artykuł: lokalne dopasowywanie szerokości jądra
  - Potrzeba takiego dopasowywania może być zauważona na podstawie poprzedniego rysunku
  - KDEC: dopasowywanie globalne
- Artykuł: musimy z góry wiedzieć ile jest obszarów dominacji szerokości jąder
- Artykuł: uczenie GMM za pomocą EM może utknąć w minimum lokalnym

Artykuł: KDE i Boosting

- Artykuł [Marzio05b]: Marco Di Marzio, Charles C. Taylor, „On boosting kernel density methods for multivariate data: density estimation and classification”, Statistical Methods & Applications, 2005
- Pomysł: Zastosowanie alg. Meta-uczącego się (Boosting) do estymacji gęstości i klasyfikacji. Modelami poziomu 0 są KDE. Modelem poziomu 1 jest Boosting.

Artykuł: KDE i Boosting cd.

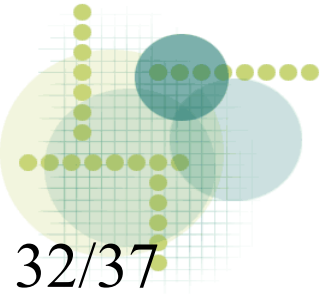
- Artykuły pokrewne:
  - Pomysł zaproponowany w [Marzio04]
  - W [Marzio05a] używa się KDE i Boosting (w wersji AdaBoost) do klasyfikacji.
    - Rozważany jest przypadek 1D z 2 klasami.
    - Przedstawiona jest analiza teoretyczna i wyniki eksperymentalne

Artykuł: KDE i Boosting cd.

- Proponują algorytm BoostKDE (do estymacji gęstości)
  - Idea: w kolejnych iteracjach boosting-u przypisujemy różne wagi każdemu z przykładów ze zbioru uczącego.
- Proponują algorytm BoostKDC (do klasyfikacji):
  - Idea: w zależności od pewnego błędu związanego z klasyfikacją przypisujemy różne wagi każdemu z przykładów ze zbioru uczącego.

Artykuł: KDE i Boosting - porównanie

- Artykuł: różna waga jest przypisywana różnym przykładom dla różnych KDE
  - KDEC: waga przykładów jest taka sama, za to dobierane są szerokości jąder

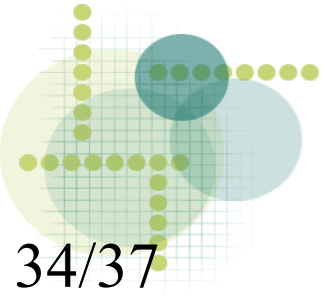


Inne artykuły

- Artykuł [Ormoneit98]: stosują uśrednianie (ensemble averaging) do estymacji gęstości. Modelami poziomu 0 są gęstości GMM. Każdej estymacji nadają taką samą wagę. Stosują 3 podejścia do generowania różnych GMM, które potem są uśrednianie:
  - Algorytm EM kończy w różnych minimach lokalnych,
  - Dane dla każdego GMM stanowią 70% danych oryginalnych is są wybierane poprzez losowanie bez powtórzeń
  - Dane dla każdego GMM wybieramy przykłady przez losowanie z powtórzeniami (Bagging)

Inne artykuły cd.

- Artykuł [Cooley98]: proponują klasyfikację za pomocą KDE, gdzie macierze kowariancji rozkładów normalnych dla każdej z klas są dobierane oddzielnie a ramach klasy są identyczne.
  - Algorytm jest przeznaczony szczególnie dla danych o dużej liczbie wymiarów
- Artykuł [Ridgeway02]: używają Boosting i Bagging do estymacji gęstości. Używają metody EM do maksymalizowania wiarogodności danych uczących.
  - Nie jest bezpośrednio optymalizowana jakość klasyfikacji.



Literatura

[Bishop95] Christopher Bishop, „Nerual Networks for pattern recognition”, p.34, 1995

[Cooley98] Craig A. Cooley, Steven N. MacEachern, „Classification via Kernel Product Estimators”, Biometrika, 1998

[Ghosh06a] Anil K. Ghosh, Probal Chaudhuri, and Debasis Sengupta, „Classification Using Kernel Density Estimates: Multi-scale Analysis and Visualization”, Technometrics, 2006

[Ghosh06b] Anil K. Ghosh, Probal Chaudhuri, C. A. Murthy, „Multiscale Classification Using Nearest Neighbor Density Estimates”, IEEE Transactions on Systems, Man, and Cybernetics, 2006

[Hastie01] Hastie, Tibshirani, Friedman, „The elements of statistical learning”, 2001

[Hoti04] Fabian Hoti, Lasse Holmstrom, „A semiparametric density estimation approach to pattern classification”, Pattern Recognition, 2004

[Hwang94] Jenq-Neng Hwang, Shyh-Rong Lay, Alan Lippman, „Nonparametric multivariate density estimation – a comparative study”, IEEE transactions on signal processing, 1994

[Marchette96] David J. Marchette, Carey E. Priebe, George W. Rogers, Jeffrey L. Solka, „Filtered Kernel Density Estimation”, Computational Statistics, 1996

[Marzio04] Marco Di Marzio, Charles C. Taylor, „Density Estimates: a Bias Reduction Technique?”, Biometrika, 2004

Literatura cd.

[Marzio05a] Marco Di Marzio, Charles C. Taylor, „Kernel density classification and boosting: an L_2 analysis”, Statistics and computing 2005

[Marzio05b] Marco Di Marzio, Charles C. Taylor, „On boosting kernel density methods for multivariate data: density estimation and classification”, Statistical Methods & Applications, 2005

[Ormoneit98] Dirk Ormoneit, Volker Tresp, „Averaging, Maximum Penalized Likelihood and Bayesian Estimation for Improving Gaussian Mixture Probability Density Estimates”, IEEE Transactions on Neural Networks, 1998

[Ridgeway02] Greg Ridgeway, „Looking for lumps: boosting and bagging for density estimation”, Computational Statistics & Data Analysis, 2002

[Scott04] David W. Scott, Stephan R. Sain, „Multidimensional Density Estimation”, Handbook of statistics vol. 24, 2004

[Shawe-Taylor07] J. Shawe-Taylor, A. Dolia, „A framework for probability density estimation”, Proceedings of International Workshop on Artificial Intelligence and Statistics, 2007.

[Smyth99] Padhraic Smyth, David Wolpert, „Linearly Combining Density Estimators via Stacking”, Machine Learning, 1999

Dziękuję za uwagę!