Klasyfikacja sygnału EKG z wykorzystaniem algorytmów kNN i eNN

Krzysztof Mazur,
Wojciech Gumuła

11 grudnia 2016

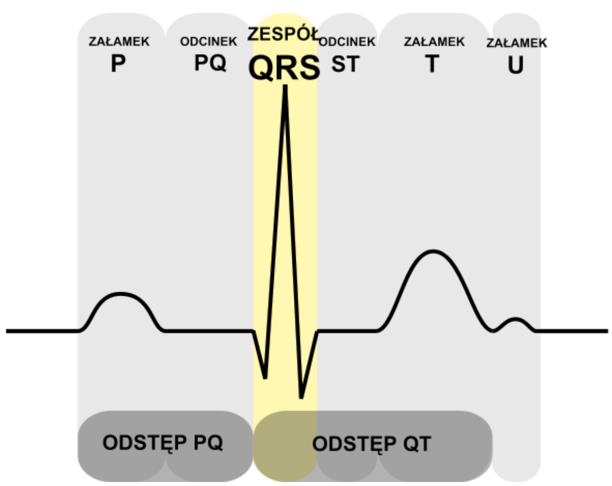
Spis treści

Rozdział 1

Wprowadzenie

Badanie elektrokardiograficzne, dzięki swej dostępności i łatwości wykonania stanowi jedno z najczęściej wykorzystywanych metod rozpoznawania zaburzeń w pracy serca. Uzyskiwany przy jego pomocy sygnał EKG dostarcza informacji o elektrycznej aktywności mięśnia sercowego, jako różnicę potencjałów pomiędzy dwoma elektrodami.

Jednym z najbardziej charakterystycznych elementów typowego sygnału EKG są zespoły QRS. Jest to układ trzech załamków opisujących proces depolaryzacji mięśnia. Ideowy schemat EKG, wraz z kompleksem QRS przedstawiono na rysunku 1.1.



Rysunek 1.1: Ideowy schemat sygnału EKG. Źródło: [3]

Na podstawie kształtu zespołu QRS diagnozować można szereg dysfunkcji serca. Wykorzystanie w tym celu zautomatyzowanych procedur diagnostycznych pozwala na zwiększenie prawdopodobieństwa wykrycia nieprawidłowości i przyśpiesza proces badania.

Klasyfikacja zespołów QRS pozwala na wyodrębnienie grup kompleksów o podobnych parametrach. Wykorzystać w tym celu można bazę *MIT-DB*

[5], zawierającą kilkadziesiąt sygnałów referencyjnych, wraz z precyzyjnym opisem stanu pacjenta i wykrytych nieprawidłowości.

Rozdział 2

Proponowane rozwiązanie

Autorzy badali możliwość wykorzystania algorytmu kNN (*k Nearest Neighbours*) oraz jego rozszerzonej wersji, eNN (*extended Nearest Neighbours*) do zaprojektowania zautomatyzowanego procesu klasyfikacji zespołów QRS do predefiniowanych grup. Opis algorytmów przedstawiono w rozdziałach 3 i 4.

Metody klasy *NN* wymagają przedstawienia danych wejściowych w postaci wektora cech o ustalonym wymiarze. Zdecydowano się wykorzystać w tym celu dane dostępne w opracowaniu [6]. Uwzględniono przy tym klasyfikację w zbiorze trzech klas, zaproponowaną przez autorów, a także w zbiorze uwzględniającym wszystkie klasy definiowane przez autorów bazy *MIT-DB*.

Dane wejściowe opisywane są wektorem składającym się z osiemnastu cech. Wektor zawiera informacje na temat chwil wystąpienia kolejnych elementów kompleksu QRS a także wartości wartość sygnału w istotnych chwilach. Jedna z kolumn - chwila wystąpienia załamka R - związana jest jednoznacznie z badanym sygnałem EKG i nie pozwala na klasyfikację w uogólnionym zbiorze danych, z tego powodu jest ignorowana w zaprojektowanym rozwiązaniu.

Zaprojektowano referencyjną implementację algorytmów *kNN* oraz *eNN* w oprogramowaniu *Matlab*, a po potwierdzeniu jej poprawności, zgodną z nią implementację w języku *C++*. Wykorzystano również bibliotekę *Eigen* [7], pozwalającą na optymalizację operacji matematycznych na macierzach i wektorach.

Cykl działania aplikacji podzielić można na dwa etapy.

1. Proces uczenia.

   Program uczony jest przy użyciu wybranego zbioru danych wejściowych wraz z poprawną klasyfikacją każdego wektora. Dane te są zapamiętywane i wykorzystywane w kolejnym etapie pracy.

2. Klasyfikacja danych wejściowych.

   Aplikacja pozwala na klasyfikację dowolnej liczby wektorów danych, porównując je z posiadanym zbiorem referencyjnym. Wyjściem programu na wejście zawierające jeden wektor testowy jest klasa, do której został on przypisany.

Test poprawności działania implementacji wymagał podzielenia znanego zbioru danych na podzbiory - uczący i testowy, wraz ze związanymi z nimi klasami. Przyjęto podział w stosunku dwa do jednego. Działanie algorytmu badane było niezależnie dla każdego pliku wejściowego.

Rozdział 3

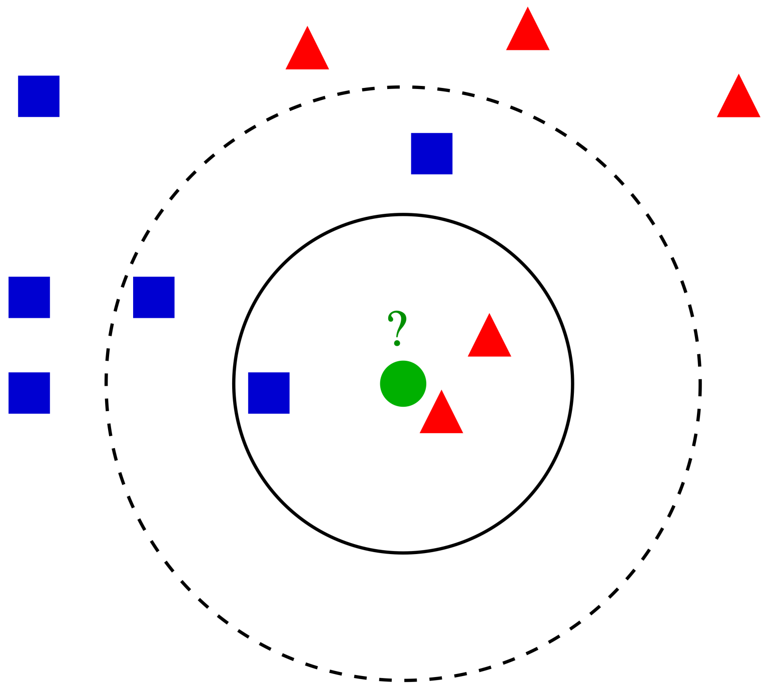
Algorytm kNN

Algorytm kNN (ang. k Nearest Neighbours) jest jednym z najprostszych algorytmów klasyfikacyjnych, jak również jednym z najpopularniejszych ze względu na prostotę implementacji oraz dobre wyniki obliczeń. Pierwszym krokiem algorytmu jest wyznaczenie odległości pomiędzy próbką ze zbioru testowego, oraz wszystkimi próbkami ze zbioru uczącego. Najczęściej stosowaną metodą wyznaczania odległości jest metryka euklidesowa dana poniższym wzorem, gdzie N to szerokość wektora dla każdej próbki:

$$d(x,y) = \sum_{i=1}^{N} \sqrt{(x_i - y_i)^2}. \tag{3.1}$$

Następnie ze zbioru K najbliższych sąsiadów wybierana jest najczęściej występująca klasa obiektów, a rozważany przypadek klasyfikowany jest jako obiekt tej klasy. Dobór optymalnego parametru K odbywa się zazwyczaj na drodze eksperymentów stosując nieparzyste wartości $K = 1, 3, 5...$. Proces uczenia jest bardzo prosty i ogranicza się do zapamiętania wszystkich próbek ze zbioru uczącego. Konieczność wyznaczenia odległości pomiędzy klasyfikowaną próbką a wszystkimi próbkami ze zbioru uczącego wymusza rozważne przygotowanie tego ostatniego. Zawarte w nim wartości powinny dobrze definiować całą przestrzeń jednocześnie minimalizując rozmiar w celu przyspieszenia działania algorytmu. Poza klasyfikacją algorytm kNN wykorzystywany jest także do klasteryzacji, regresji, estymowania funkcji gęstości. W 2006 roku algorytm został zaliczony do 10 najpopularniejszych algorytmów w dziedzinie eksploracji danych podczas konferencji IEEE w Hong Kongu.

Zasadę działania algorytmu na przykładzie przedstawiono na rysunku 3.1.



Rysunek 3.1: Schematyczne przedstawienie klasyfikacji dwuwymiarowego wektora danych. Źródło [4]

Zadaniem klasyfikacji jest ustalenie przynależności wartości wejściowej - koła, do jednej z dwóch grup elementów - kwadratów i trójkątów. Przyjmując wartość parametru $K = 3$ (linia ciągła na rysunku), wartość przypisana zostanie do grupy zawierającej trójkąty. Zakładając $K = 5$ wynik będzie przeciwny i element zostanie powiązany z grupą zawierającą kwadraty.

Dobór wartości parametru K dającej najlepsze wyniki wymaga więc znajomości badanego zjawiska bądź przeprowadzenia doświadczeń na reprezentatywnym zbiorze testowym.

Rozdział 4

Algorytm eNN

4.1 Podstawowa wersja algorytmu

Algorytm kNN pomimo wielu swoich zalet ma także kilka wad, które w niektórych przypadkach mogą niekorzystnie wpływać na wyniki działania. Jednym z charakterystycznych problemów jest przypadek kiedy rozkłady gęstości poszczególnych klas nie są równe. W takiej sytuacji elementy klasy o większym rozkładzie gęstości przeważają nad elementami pozostałych klas na danym obszarze. Dla algorytmu większość najbliższych sąsiadów będzie pochodziła z dominującej klasy co może zakłócać poprawność działania.

Jako rozwinięcie algorytmu kNN został stworzony algorytm eNN (*ang. Extended Nearest Neighbours*) [1]. Zaproponowana metoda podczas klasyfikacji bierze pod uwagę nie tylko k najbliższych sąsiadów klasyfikowanego obiektu, ale również ich otoczenie.

Dla uproszczenia opis algorytmu zostanie przeprowadzony dla przypadku klasyfikacji pomiędzy dwoma klasami. Uogólnienie algorytmu dla dowolnej ilości klas sprowadza się jedynie do większej złożoności obliczeniowej. Dla każdego z k najbliższych sąsiadów próbki ze zbioru testowego wyznaczana jest funkcja statystyczna T opisująca otoczenie w jakim się znajduje. Funkcja T wyznaczana jest według wzoru:

$$T_i = \frac{1}{n_i k} \sum_{x \in S_i} \sum_{r=1}^{k} I_r(x, S \in (S_1 \cup S_2)) \tag{4.1}$$

gdzie S_1 i S_2 są zbiorami próbek należących odpowiednio do klas 1 i 2, k jest zdefiniowaną ilością najbliższych sąsiadów a I_r dane jest wzorem:

$$I_r(x,S) = \begin{cases} 1 & dla\ x \in S_i \wedge NN_r(x,S) \in S_i \\ 0 & w\ pozostaych\ przypadkach \end{cases} \tag{4.2}$$

Funkcja I_r przyjmuje wartość 1 jeżeli próbka x i jej r-sąsiad należą do tej samej klasy, natomiast wartość 0 przyjmuje w pozostałych przypadkach. Funkcja T_i przyjmuje wartości z zakresu [0,1]. Im większa wartość T_i tym więcej próbek tej samej klasy znajduje się w otoczeniu badanej próbki. Małe wartości oznaczają małą ilość próbek tej samej klasy.

Mając nową, niesklasyfikowaną próbkę, w kolejnych krokach zostaje przypisana do poszczególnych klas a następnie dla każdego przypadku wyznaczona zostaje wartość funkcji T_i^j według wzoru:

$$T_i^j = \frac{1}{n_i^j k} \sum_{x \in S'_{i,j}} \sum_{r=1}^{k} I_r(x, S' \in (S_1 \cup S_2 \cup Z)) \tag{4.3}$$

gdzie j oznacza klasę do której została przypisana próbka Z. W rozważanym przypadku, gdy pod uwagę brane są dwie klasy otrzymujemy cztery wyniki $T_1^1, T_2^1, T_1^2\ oraz\ T_2^2$. Próbka Z zostaje zakwalifikowana zgodnie ze wzorem:

$$f_{ENN} = arg_{j \in 1,2} max \sum_{i=1}^{2} T_i^j \tag{4.4}$$

Powyższy wzór w przypadku gdy w zbiorze znajduje się N klas występuje w postaci:

$$f_{ENN} = arg_{j \in 1,2,..,N} max \sum_{i=1}^{N} T_i^j \tag{4.5}$$

Klasyfikacja próbki do odpowiedniej klasy odbywa się poprzez wybór maksymalnej wartości funkcji T_i spośród wyznaczonych metodą iteracyjną wartości dla kolejnych klas.

4.2 Alternatywna wersja algorytmu

Analizując podstawową wersję algorytmu eNN można zaobserwować konieczność wyznaczenia wartości T_i^j dla każdej kolejnej klasyfikowanej próbki. Wiąże się to ze wzrostem złożoności obliczeniowej co nie jest pożądane w przypadku dużych zbiorów testowych. Rozwiązaniem tego problemu jest równoważna wersja algorytmu eNN, która została przedstawiona w tym paragrafie.

W rozważanej wersji algorytmu pod uwagę brane jest nie tylko kto znajduje się w zbiorze najbliższych sąsiadów klasyfikowanej próbki Z ale także które próbki ze zbioru uczącego biorą pod uwagę próbkę Z jako jednego z k najbliższych sąsiadów. W tym celu próbka Z zostaje przypisana kolejno do wszystkich klas. W każdym przypadku zostają wyznaczone wartości Δn_i^j gdzie j jest klasą do której została przypisana próbka Z, natomiast i jest klasą dla której obliczana jest zmiana ilości sąsiadów tej samej klasy. W przypadku gdy i = j zliczana jest ilość próbek klasy i dla której wzrosła ilość sąsiadów należących do tej samej klasy. W przypadku gdy $i \neq j$ wyznaczana jest ilość próbek klasy i dla których zmalała ilość sąsiadów należących do tej samej klasy.

Dla wyznaczonych wartości n_i^j klasyfikacja próbki Z realizowana jest zgodnie z zależnością:

$$f_{ENN} = arg_{j \in 1,2,..,N} max \left\{ \left(\frac{\Delta n_i^j + k_j - kT_i}{(n_i+1)k} \right) \right\}_{i=j} - \sum_{i \neq j}^{N} \frac{\Delta n_i^j}{n_i k} \qquad (4.6)$$

gdzie k jest zdefiniowaną ilością najbliższych sąsiadów, n_i jest ilością próbek należących do klasy i, a k_i jest ilością najbliższych sąsiadów należących do klasy i.

Obie wersje algorytmu są równoważne, jednak w przypadku drugiej wersji na początku działania algorytmu można wyznaczyć wektor wartości T_i a podczas klasyfikacji kolejnych próbek ograniczyć się do wyznaczenia zmian w otoczeniach k najbliższych sąsiadów próbki Z, dla kolejnych przypisań próbki Z do odpowiednich klas.

Bibliografia

[1] Bo Tang i Haibo He. *Enn: Extended Nearest Neighbor Method for Pattern Recognition*. IEEE Computational intelligence magazine. 2015

[2] Haibo He. *Referencyjna implementacja algorytmu eNN*.
`http://www.ele.uri.edu/faculty/he/research/ENN/ENN.html`

[3] Wikipedia. *QRS complex*.
`https://en.wikipedia.org/wiki/QRS_complex`

[4] Wikipedia. *k-nearest neighbors algorithm*.
`https://en.wikipedia.org/wiki/K-nearest_neighbors_algorithm`

[5] *MIT-BIH Arrhythmia Database*.
`https://www.physionet.org/physiobank/database/mitdb/`

[6] Michał Ciszewski, Łukasz Dudek i Krystian Mucha. *Moduł HeartClass*.

[7] *Eigen Library*.
`http://http://eigen.tuxfamily.org/`