

Laboratorium Rozpoznawania Obrazów – Ćwiczenie #3 & #4

Rozpoznawanie cyfr pisanych ręcznie

Termin oddawania: **9.04.2017, 12.04.2017, 19.04.2017, 7.05.2017**

W tym ćwiczeniu Państwa zadaniem będzie rozpoznawanie cyfr pisanych ręcznie za pomocą zestawu klasyfikatorów liniowych. Do pierwszej fazy zadania – przygotowania klasyfikatorów – wykorzystamy dane z bazy danych MNIST (<http://yann.lecun.com/exdb/mnist>). Zbiory uczący i testowy można pobrać spod podanego wyżej adresu. Dodatkowo są dostępne na serwerze Galera.

Warto wiedzieć, że obrazy cyfr wydzielone z zeskanowanej strony są normalizowane w następujący sposób:

1. Prostokąt zawierający czarno-biały obraz znaku zeskanowanego w rozdzielczości 300 dpi jest skalowany proporcjonalnie do prostokąta o większym z wymiarów równym 20. W trakcie skalowania obraz jest zamieniany na skalę szarości (proporcjonalnie do liczby oryginalnych pikseli pierwszego planu przypadających na jeden piksel obrazu po przeskalowaniu).
2. Jest wyznaczany środek ciężkości przeskalowanego znaku, a znak jest umieszczany w obrazie 28x28 pikseli tak, żeby środek ciężkości znalazł się na środku tego większego obrazu.

Drugi zestaw znaków został poddany dodatkowo usuwaniu przekosu.

Łączy do plików na serwerze Galera są następujące:

<http://galeranew.ii.pw.edu.pl/~rkz/mnist.zip> (oryginalne dane MNIST)

Państwa zadaniem, jest przygotowanie klasyfikatora korzystającego z **klasyfikatorów liniowych** rozróżniających konkretne cyfry. Oprócz jakości klasyfikacji na zbiorze testowym należy podać **macierz pomyłek**. Dodatkowe zadanie polega na poprawieniu jakości klasyfikacji w stosunku do klasyfikacji zrealizowanej przez zwykłe głosowanie klasyfikatorów elementarnych. Warto zwrócić uwagę, że znaczny wpływ na jakość klasyfikacji ma ograniczenie zestawu klas, dla których jest przeprowadzane głosowanie.

Rozwiązaniem referencyjnym, jest klasyfikator głosujący (wykorzystujący głosy 45 klasyfikatorów liniowych *one vs. one*) wskazujący jako wynik konkretną cyfrę, o ile zostało oddanych na nią 9 głosów. W pozostałych przypadkach jest podejmowana decyzja wymijająca. W testach użyłem 40 składowych głównych. Wyniki klasyfikacji podsumowuje poniższa tabela (choć ciekawy wgląd w klasyfikację może dać analiza macierzy pomyłek).

	Zbiór uczący MNIST			Zbiór testowy MNIST		
	OK.	Błąd	Odrzucenie	OK.	Błąd	Odrzucenie
Jakość klasyfikacji	91.34%	5.72%	2.94%	91.55%	5.49%	2.96%

Zadanie można podzielić na kilka części:

1. Przygotowanie podstawowego algorytmu wyznaczającego parametry płaszczyzny decyzyjnej dla zadanego zbioru uczącego, zawierającego dwie klasy. W zupełności wystarczy użycie tutaj algorytmu uczenia perceptronu, ale możecie Państwo poeksperymentować i z innymi metodami wyznaczania płaszczyzny decyzyjnej.
2. Testy algorytmu dla wielowymiarowych danych cyfr. Przeprowadzenie tego testu ma na celu sprawdzenie wydajności algorytmu oraz jego zachowania w przestrzeni

wielowymiarowej. Dane źródłowe mogą być użyte bezpośrednio (tzn. każdy piksel oryginalnych obrazów jest traktowany jako cecha), lub po redukcji wymiarowości algorytmem PCA.

3. Rozwiązanie „kanoniczne” polega na przygotowaniu 45 klasyfikatorów – po jednym dla każdej pary cyfr – i przeprowadzeniu głosowania klasyfikatorów na poszczególne cyfry.

I tu pojawiają się pierwsze ciekawe pytania: Czy wystarczy wybrać klasę o największej liczbie głosów? Co zrobić, jeśli więcej niż jedna klasa dostała maksymalną liczbę głosów? Czy klasyfikator – wzorem rozwiązania referencyjnego – powinien dawać decyzję wymijającą?

Metodę głosowania, którą przyjmiecie Państwo w tym punkcie, będziecie również stosować w następnym kroku. Proszę zatem pamiętać, żeby nie była przywiązana do konkretnej liczby cyfr (w kanonicznym macie 10 cyfr; w następnym od 2 do 4).

4. Ostatni krok, to ograniczenie liczby klas wchodzących do ostatniej fazy głosowania. Co ważne, w tej ostatniej fazie należy użyć klasyfikatorów z rozwiązania kanonicznego – chcemy wszak sprawdzić, jaki wpływ na jakość klasyfikacji ma ograniczenie liczby klas!

Do przeprowadzenia wstępnej klasyfikacji również użyjcie Państwo klasyfikatorów liniowych. Zadanie nie jest proste bezpośrednio w przestrzeni cech. Można próbować uczenia zestawu klasyfikatorów liniowych, a ostateczną decyzję podejmować zwykłym głosowaniem większościowym. Kluczowa jest tutaj różnorodność klasyfikatorów: można próbować uczyć klasyfikatory tylko na części próbek ze zbioru uczącego (jakiej części to otwarte pytanie), używać różnych zestawów cech (składowych PCA lub pikseli).

Rozwiązaniem, które nieco podważa ostatni etap klasyfikacji, ale jest akceptowalne w tym ćwiczeniu jest użycie maszyny wektorów nośnych (tylko z jądrem liniowym i tylko do rozdzielania cyfr na grupy; dobrym wyborem jest biblioteka libsvm, która ma wersję bez problemów współpracującą z Octave). W tym przypadku proszę odnotować w sprawozdaniu liczbę wektorów nośnych dla każdego klasyfikatora.

W sprawozdaniu proszę zamieścić:

1. Opis metody klasyfikacji znaków przy użyciu klasyfikatorów liniowych.
2. Opis algorytmu wyznaczania parametrów płaszczyzny decyzyjnej.
3. Dane dotyczące jakości klasyfikacji każdego z wykorzystywanych klasyfikatorów liniowych.
4. Dane dotyczące jakości klasyfikacji cyfr, z wnioskami wynikającymi z analizy **macierzy pomyłek**.
5. Opis niestandardowego podejmowania decyzji klasyfikacyjnej i porównanie jej wyników z rozwiązaniem kanonicznym.

W przypadku podziału na grupy trzeba podać:

Uzasadnienie dla zastosowanego podziału cyfr na grupy.

Dane dotyczące jakości tego wstępnego podziału na grupy cyfr (także z macierzą pomyłek).

Dane całościowe klasyfikacji i porównanie ich z rozwiązaniem kanonicznym.

Do pakietu sprawozdanie + kod proszę **nie dołączać danych** (koszt dołączenia danych wynosi 2 punkty)!