

Laboratorium Rozpoznawania Obrazów – Ćwiczenie #3 & #4

Rozpoznawanie cyfr pisanych ręcznie

Termin oddawania: **20.04.2015, 23.04.2015**

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

<http://galera.ii.pw.edu.pl/~rkz/reference.zip> (rozwiązanie referencyjne - warto w nim sprawdzić, jak odczytywać dane – lub po prostu przejąć z niego funkcję odczytu danych)

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 zaproponowaniu innego (bardziej wyrafinowanego?) sposobu wyznaczania decyzji klasyfikatora, niż zwykle głosowanie klasyfikatorów elementarnych.

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. Ten wstępny krok najlepiej przeprowadzić na zbiorze dwuwymiarowym, który będzie łatwo

zwizualizować i pokazać wyznaczone rozwiązanie. Oprócz przejrzenia materiałów z wykładu wartościowe będzie zajrzenie do *Numerical Recipes in C* (rozdział 10, „Minimization and Maximization of Functions”).

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?

4. Ostatni krok – którym wg mnie warto zająć się kiedy klasyfikator kanoniczny będzie gotowy – jest podejście do niestandardowego wykorzystania klasyfikatorów elementarnych.

Kilka pomysłów, które można przynajmniej rozważyć:

Wykorzystać informacje o błędach na zbiorze uczącym klasyfikatorów elementarnych (głosowanie ważone).

Uwzględnić odległość klasyfikowanej próbki od płaszczyzny decyzyjnej (tu warto pamiętać, że odległości muszą być porównywalne ze sobą, czyli trzeba znormalizować długość wektorów normalnych hiperpłaszczyzn; warto na pewno sprawdzić, jakie odległości mają próbki źle klasyfikowane).

Bardziej precyzyjnie rozdzielić klasy cyfr, tak żeby były lepiej separowalne liniowo (czyli przeprowadzić mniej lub bardziej wyrafinowane grupowanie).

W inny sposób (przy użyciu innego klasyfikatora) ograniczyć liczbę głosujących klasyfikatorów cyfra do cyfry.

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.