

**PROJEKT „METODY ZŁOŻONEGO ROZPOZNAWANIA”
studia niestacjonarne AiR II rok**

Numer zajęć	Zadania do wykonania
1	Przekazanie wymagań oraz zapoznanie ze sposobem oceny
2	Zgłoszenie tematu projektu
3-4	Przeprowadzenie studiów literaturowych. Przekazanie opracowanie na wybrany temat.
5-6	Przedstawienie założeń projektowych oraz plany eksperymentu.
7-10	Oddanie projektu

Projekt końcowy powinien zawierać:

1. Przegląd dostępnej literatury oraz opracowanie teoretyczne na wybrany temat.
2. Kompletny projekt aplikacji wykonany w wybranej metodologii projektowania.
3. Jasno sformułowany cel eksperymentu.
4. Plan eksperymentu
5. Wyniki.
6. Wnioski

Wykonanie projektu polega nie tylko na wykonaniu aplikacji oraz testów komputerowych, ale także (a właściwie przede wszystkim) na opracowaniu wyników i wyciągnięciu odpowiednich wniosków.

Termin zaliczenia końcowego projektu – 10 zjazd. Po tych terminach projektów nie przyjmuje.

Ocena końcowa będzie średnią ważoną i będą brane pod uwagę następujące elementy:

- | | |
|---|-----|
| 1. Wybranie oryginalnego tematu - | 5% |
| 2. Studia literaturowe - | 20% |
| 3. Prawidłowo sprecyzowane założenia projektowe | 10% |
| 4. Projekt końcowy | 65% |

Nieterminowe zgłoszenie tematu, oddania któregoś opracowania po terminie wiąże się z oceną niedostateczną za powyższy element. Będę jedynie honorował *usprawiedliwienia* lekarskie przekazane na następnych zajęciach (jeżeli oczywiście wszyscy członkowie grupy będą chorzy, jak nie to zdrowy powinien się stawić w wymienionym terminie).

Poniżej podaję przykładowe tematy projektów, ale mogą Państwo zgłaszać inne (oczywiście o podobnym stopniu skomplikowania)

1. Ilustracja klasyfikacji za pomocą algorytmów:

- Parzena
- K-najbliższych sąsiadów
- K-tego najbliższego sąsiada

Badania wykonać dla wybranych 3 baz benchmarkowych z UCI ML Repository (podać jakość rozpoznawania metodą walidacji krzyżowej, a także wariancje odpowiedzi).

2. Ilustracja klasyfikacji za pomocą algorytmów:

- Parzena
- K-najbliższych sąsiadów
- K-tego najbliższego sąsiada

Dla problemu dwuklasowego, z dwoma cechami o rozkładach normalnych (tylko ilustracja graficzna), a także dla zbioru o rozkładzie „bannana” (podać jakość rozpoznawania metodą walidacji krzyżowej, a także wariancje odpowiedzi + ilustracja graficzna).

3. Ilustracja klasyfikacji za pomocą algorytmów:

- Funkcji potencjałowych
- Liniowej dyskryminacji Fishera

Badania wykonać dla wybranych 3 baz z UCI ML Repository (podać jakość rozpoznawania metodą walidacji krzyżowej, a także wariancje odpowiedzi).

4. Ilustracja klasyfikacji za pomocą algorytmów:

- Funkcji potencjałowych
- Liniowej dyskryminacji Fishera

dla problemu dwuklasowego, z dwoma cechami o rozkładach normalnych (tylko ilustracja graficzna), a także dla zbioru o rozkładzie „bannana” (podać jakość rozpoznawania metodą walidacji krzyżowej, a także wariancje odpowiedzi + ilustracja graficzna).

5. Jakość rozpoznawania klasyfikatorów po przejściu przez procedurę *baggingu* z jakością klasyfikatorów pierwotnych (proponuję k-NN). Badania wykonać dla różnych długości ciągu uczącego. Do badań wykorzystać wybrać co najmniej 3 bazy z UCI ML Repository (podać jakość rozpoznawania metodą walidacji krzyżowej, a także wariancje odpowiedzi).

6. Porównać jakość klasyfikatorów opartych o ideę vottingu. Proszę wykonać badania dla niewielkiej grupy klasyfikatorów uczonych na nielicznych (rozłącznych) podzbiorach danych z UCI ML Repository (podać jakość rozpoznawania metodą walidacji krzyżowej, a także wariancje odpowiedzi). Porównać łączenie klasyfikatorów homogenicznych i heterogenicznych.

7. Ilustracja zjawiska *overfittingu* dla klasyfikatorów drzewiastych. Do badań wykorzystać wybrane bazy z UCI ML Repository.

8. Porównanie jakości wybranych algorytmów kombinowanych dla zbioru „bannana”. Porównać idee *baggingu*, prostego głosowania oraz połączenia klasyfikatorów szeregowo, równoległe i równoległe-szeregowo. Proponuję jako bazę wybrać klasyfikator k-NN i wykonać badania dla różnej długości ciągu uczącego.

Proponuję wykorzystać Matlab i toolboxa PRTools

<http://www.prtools.org>

UCI Machine Learning Repository

<http://www.ics.uci.edu/~mllearn/MLRepository.html>

	Nazwa bazy	Przykładów	Atrybutów	Klas
1	annealing	798	38	5
2	audiology	200	70	24
3	autos	205	25	6
4	balance-scale	625	4	3
5	breast-cancer	286	9	2
6	breast-cancer-wisconsin	699	10	2
7	credit-screening	690	15	2
8	flags	194	29	8
9	glass	214	10	6
10	hepatitis	155	19	2
11	ionosphere	351	34	2
12	iris	150	4	3
13	letter-recognition	20000	16	26
14	lymphography	148	18	4
15	mushroom	8124	22	2
16	pima-indians-diabetes	768	8	2
17	primary-tumor	339	17	21
18	tic-tac-toe	958	9	2
19	voting-records	435	16	2
20	wine	178	13	3