



KARTA OPISU PRZEDMIOTU - SYLABUS

Nazwa przedmiotu

Metody uczenia maszynowego

Przedmiot

Kierunek studiów

Rok/semestr

Matematyka w technice

1/2

Studia w zakresie (specjalność)

Profil studiów

Programowanie w technice

ogólnoakademicki

Poziom studiów

Język oferowanego przedmiotu

drugiego stopnia

polski

Forma studiów

Wymagalność

stacjonarne

obieralny

Liczba godzin

Wykład

Laboratoria

Inne (np. online)

30

15

Ćwiczenia

Projekty/seminaria

Liczba punktów ECTS

3

Wykładowcy

Odpowiedzialny za przedmiot/wykładowca:

Odpowiedzialny za przedmiot/wykładowca:

dr inż. Marek Kraft

Wymagania wstępne

ma pogłębioną i podbudowaną teoretycznie wiedzę z informatyki, w tym z metod numerycznych; zna szczególnie co najmniej jeden pakiet oprogramowania lub język programowania

potrafi skonstruować algorytm rozwiązania złożonego zadania inżynierskiego lub prostego problemu badawczego oraz zaimplementować i przetestować go w wybranym środowisku programistycznym

potrafi dobrać odpowiednie źródła wiedzy i pozyskać z nich niezbędne informacje, dokonać krytycznej analizy i oceny rozwiązań złożonych i nietypowych zadań inżynierskich lub prostych problemów badawczych oraz zaproponować ich ulepszenie

Cel przedmiotu

Celem przedmiotu jest zapoznanie studentów z zagadnieniami i algorytmami z dziedziny uczenia maszynowego oraz wykorzystania zdobytej wiedzy do rozwiązywania praktycznych zadań.

Przedmiotowe efekty uczenia się

Wiedza



ma pogłębioną i podbudowaną teoretycznie wiedzę z informatyki, w tym z metod numerycznych; zna szczególnie co najmniej jeden pakiet oprogramowania lub język programowania

zna i rozumie wpływ matematyki na postęp nauki

Umiejętności

potrafi wykorzystywać techniki, narzędzia i metody matematyczne, w tym numeryczne lub optymalizacyjne do rozwiązywania zaawansowanych zadań inżynierskich lub prostych problemów badawczych

potrafi skonstruować algorytm rozwiązania złożonego zadania inżynierskiego lub prostego problemu badawczego oraz zaimplementować i przetestować go w wybranym środowisku programistycznym

Kompetencje społeczne

jest świadomy roli i znaczenia wiedzy w rozwiązywaniu problemów o charakterze poznawczym oraz praktycznym, typowych dla zawodów i miejsc pracy właściwych dla absolwentów studiowanego kierunku; ma świadomość konieczności pogłębiania i poszerzania wiedzy

jest gotów do myślenia i działania w sposób kreatywny i przedsiębiorczy, uwzględniając bezpieczeństwo, ergonomię pracy i jej ekonomiczne aspekty; jest świadomy konieczności inspirowania i organizowania działania na rzecz interesu publicznego oraz odpowiedzialności za efekty pracy zespołu, jak i poszczególnych jego uczestników; wykazuje gotowość do wypełniania społecznych zobowiązań wynikających z charakteru pracy typowej dla absolwentów studiowanego kierunku

Metody weryfikacji efektów uczenia się i kryteria oceny

Efekty uczenia się przedstawione wyżej weryfikowane są w następujący sposób:

Wykład - test zaliczeniowy przeprowadzany na platformie Moodle. Laboratoria - projekt zaliczeniowy.

Treści programowe

Wykład:

Definicja uczenia maszynowego i różnice między uczeniem maszynowym a tradycyjnym programowaniem.

Uczenie maszynowe nadzorowane, nienadzorowane, uczenie ze wzmocnieniem.

Ocena jakości działania metod uczenia maszynowego - miary i metryki.

Rola cech w uczeniu maszynowym.

Prezentacja algorytmów uczenia maszynowego, sposobu ich działania i charakterystyki: klasyfikator Bayesa, drzewa decyzyjne, random forest, maszyny wektorów nośnych, klasteryzacja, sieci neuronowe.

Uczenie ze wzmocnieniem - algorytmy i zastosowania.

Laboratoria:

Zapoznanie z bibliotekami scikit-learn oraz TensorFlow. Implementacja wybranych algorytmów z wykorzystaniem bibliotek, ocena działania i graficzna prezentacja efektów działania algorytmów w praktycznych aplikacjach.



Metody dydaktyczne

Wykłady z prezentacjami multimedialnymi, zamieszczane następnie dodatkowo w serwisie streamingowym do późniejszego odtworzenia. Zajęcia laboratoryjne z wykorzystaniem notatników w języku Python, obejmujące implementację i testowanie wybranych algorytmów uczenia maszynowego do rozwiązywania problemów w wybranych zastosowaniach.

Literatura

Podstawowa

Sebastian Raschka, Vahid Mirjalili, Python. Uczenie maszynowe. Helion, 2019

Uzupełniająca

Bengio, Yoshua, Ian Goodfellow, and Aaron Courville. Deep learning. Vol. 1. Massachusetts, USA: MIT press, 2017.

Bilans nakładu pracy przeciętnego studenta

	Godzin	ECTS
Łączny nakład pracy	90	3,0
Zajęcia wymagające bezpośredniego kontaktu z nauczycielem	45	1,5
Praca własna studenta (studia literaturowe, przygotowanie do zajęć laboratoryjnych/ćwiczeń, przygotowanie do kolokwίων/egzaminu, wykonanie projektu) ¹	45	1,5

¹ niepotrzebne skreślić lub dopisać inne czynności