



KARTA OPISU PRZEDMIOTU - SYLABUS

Nazwa przedmiotu

Programowanie z elementami uczenia maszynowego [S1MNT1>PzEUM]

Przedmiot

Kierunek studiów

Matematyka nowoczesnych technologii

Rok/Semestr

1/2

Studia w zakresie (specjalność)

–

Profil studiów

ogólnoakademicki

Poziom studiów

pierwszego stopnia

Język oferowanego przedmiotu

polski

Forma studiów

stacjonarne

Wymagalność

obligatoryjny

Liczba godzin

Wykład

15

Laboratorium

30

Inne

0

Ćwiczenia

0

Projekty/seminaria

15

Liczba punktów ECTS

5,00

Koordynatorzy

dr Grzegorz Oleksik

grzegorz.oleksik@put.poznan.pl

dr inż. Nadiia Bashova

nadiia.bashova@put.poznan.pl

Wykładowcy

Wymagania wstępne

Student rozpoczynający ten przedmiot powinien posiadać wiedzę i umiejętności kursu Wstęp do programowania oraz Technologie informacyjne z semestru pierwszego. Powinien znać ograniczenia własnej wiedzy i rozumieć potrzebę dalszego kształcenia.

Cel przedmiotu

Celem przedmiotu jest zaznajomienie studenta z przydatnymi pakietami numerycznymi oraz związanymi z analizą danych i uczeniem maszynowym. Wykorzystane są do tego celu odpowiednie biblioteki języka Python (numpy, matplotlib, pandas, scikit-learn, TensorFlow).

Przedmiotowe efekty uczenia się

Wiedza:

- zna i rozumie w zaawansowanym stopniu narzędzia matematyki do analizy danych [K_W01(P6S_WG), K_W05(P6S_WG), K_W07(P6S_WG)];

• ma uporządkowaną i podbudowaną teoretycznie wiedzę z informatyki, w tym z metod numerycznych; zna co najmniej jeden pakiet oprogramowania lub język programowania [K_W 01(P 6S_W G), K_W 05(P 6S_W G), K_W07(P6S_WG)].

Umiejętności:

- student potrafi zastosować wiedzę teoretyczną, w szczególności z matematyki, do przetwarzania i analizy danych oraz formułowania odpowiednich wniosków [K_U04(P6S_UW), K_U05(P6S_UW), K_U06(P6S_UW)];
- student potrafi gromadzić / przetwarzać dane oraz oceniać ich jakość [K_U06(P6S_UW)];
- student potrafi skonstruować algorytm rozwiązania prostego zadania inżynierskiego oraz zaimplementować i przetestować go w wybranym środowisku programistycznym [K_U11(P6S_UW)];
- student potrafi zgodnie z ogólnymi wymogami i dokumentacją techniczną eksploatować urządzenia, narzędzia itp.; umie stosować zasady bezpieczeństwa i higieny pracy [K_U17(P6S_UU)].

Kompetencje społeczne:

- student jest gotów do wsparcia innych jednostek naukowych / przemysłu itp. w zakresie modelowania matematycznego / wnioskowania statystycznego / analizy i przetwarzania danych na rzecz środowiska społecznego [K_K01(P6S_KK), K_K02(P6S_KK)];
- student ma świadomość poziomu swojej wiedzy w odniesieniu do prowadzonych badań w naukach ścisłych i technicznych [K_K01(P6S_KK)];
- student ma świadomość pogłębiania i poszerzania wiedzy do rozwiązywania nowopowstałych problemów technicznych [K_K02(P6S_KK)].

Metody weryfikacji efektów uczenia się i kryteria oceny

Efekty uczenia się przedstawione wyżej weryfikowane są w następujący sposób:

Wykłady: wiedza nabyta w ramach wykładu jest weryfikowana w trakcie końcowego testu. Zasady punktacji są następujące:

Wynik Ocena

0% - 49% Niedostateczny
50% - 59% Dostateczny
60% - 69% Dostateczny plus
70% - 79% Dobry
80% - 89% Dobry plus
90% -100% Bardzo dobry

Laboratoria

1. W czasie semestru przewidziano dwa równo punktowane kolokwia (po 30 pkt).
2. Nie przewiduje się poprawiania kolokwiiów.
3. Student może uzyskać dodatkowe punkty za aktywność podczas zajęć nie więcej niż 10% (czyli maksymalnie 6 pkt).
4. Obecność na zajęciach laboratoryjnych jest obowiązkowa.
5. Zaliczenie co najmniej 75% zadań na zajęciach jest obowiązkowe.
6. Suma punktów uzyskana ze sprawdzianów i aktywności podczas zajęć przekłada się na ocenę końcową zgodnie z poniższą tabelą.

Wynik Ocena

0% - 49% Niedostateczny
50% - 59% Dostateczny
60% - 69% Dostateczny plus
70% - 79% Dobry
80% - 89% Dobry plus
90% -100% Bardzo dobry

7. Student, który nie uzyska oceny pozytywnej, ma prawo do kolokwium poprawkowego w sesji poprawkowej.

Projekt

1. Maksymalna liczba punktów za projekt wynosi 60 pkt.
2. Przy ocenianiu projektu będą brane pod uwagę następujące kryteria:
 - Działanie i jakość kodu
 - Metody uczenia maszynowego i analiza wyników
 - Dobór danych
 - Złożoność projektu
 - Dokumentacja
 - Prezentacja projektu
3. Student może uzyskać dodatkowe punkty za sprinty podczas zajęć — nie więcej niż 10% (czyli maksymalnie 6 pkt).
4. Suma punktów uzyskanych za projekt i aktywność podczas zajęć przekłada się na ocenę końcową zgodnie z poniższą tabelą:

Wynik Ocena

0% - 49% Niedostateczny

50% - 59% Dostateczny

60% - 69% Dostateczny plus

70% - 79% Dobry

80% - 89% Dobry plus

90% -100% Bardzo dobry

5. Student, który nie uzyska oceny pozytywnej, ma prawo do przedstawienia projektu w sesji poprawkowej.

Treści programowe

Wybrane elementy języka Python oraz wybrane biblioteki związane z analizą danych i uczeniem maszynowym.

Tematyka zajęć

Wykłady

- słowniki;
- praca na plikach (w tym obsługa plików csv, biblioteka json) ;
- obliczenia numeryczne - biblioteka numpy;
- przetwarzanie - i obróbka danych - biblioteka pandas.
- podstawy uczenia maszynowego (perceptron, typy uczenia)
- korzystanie z metod i modeli modułu Scikit-Learn oraz TensorFlow

Laboratorium:

- praca na plikach różnego typu (json, csv, excel)
- obróbka i analiza danych przy użyciu bibliotek: numpy, matplotlib, pandas, Scikit-Learn, TensorFlow

Projekt:

- postawieniu celu badań
- znalezienie odpowiednich danych
- przygotowanie danych do analizy
- przetwarzanie danych przy użyciu wybranych metod uczenia maszynowego
- używanie metod pracy zespołowej (metodyka Scrum)

Metody dydaktyczne

Wykłady:

- wykład z prezentacją uzupełniany przykładami podawanymi na tablicy;
- wykład prowadzony w sposób interaktywny z formułowaniem pytań do grupy studentów lub do wskazywanych konkretnych studentów;
- uwzględnia się aktywność studentów w czasie zajęć przy wystawianiu oceny końcowej;
- w trakcie wykładu inicjowanie dyskusji;
- teoria przedstawiana w ścisłym powiązaniu z praktyką;

- teoria przedstawiana w powiązaniu z aktualną wiedzą studentów;
- przedstawianie nowego tematu poprzedzone przypomnieniem treści powiązanych, znanych studentom z innych przedmiotów.

Laboratoria& Projekty/seminaria:

- laboratoria uzupełniane prezentacjami multimedialnymi;
- recenzowanie sprawozdań przez prowadzącego laboratoria i dyskusje nad komentarzami;
- korzystanie z narzędzi umożliwiających studentom wykonanie zadań w domu (np. oprogramowanie open source);
- demonstracje;
- praca w zespołach;
- eksperymenty obliczeniowe.

Literatura

Podstawowa (zalecane najnowsze wydania):

- Geron, A. *Uczenie Maszynowe z użyciem Scikit-Learn, Keras i TensorFlow*
- McKinney W., *Python w analizie danych. Przetwarzanie danych za pomocą pakietów Pandas i NumPy oraz środowiska IPython*, Wydawnictwo Helion, 2018;
- Gągolewski M., Bartoszek M., Cena A., *Przetwarzanie i analiza danych w języku Python*, Wydawnictwo Naukowe PWN, Warszawa, 2022;
- Grus J., *Data science od podstaw. Analiza danych w Pythonie*, Wydawnictwo Helion, 2020.

Uzupełniająca:

- Larose D. T., *Metody i modele eksploracji danych*, Wydawnictwo Naukowe PWN;
- Larose D. T., *Odkrywanie wiedzy z danych*, Wydawnictwo Naukowe PWN, Warszawa 2006;
- Morzy T., *Eksploracja danych. Metody i algorytmy*, Wydawnictwo Naukowe PWN, Warszawa 2013 .

Bilans nakładu pracy przeciętnego studenta

	Godzin	ECTS
Łączny nakład pracy	125	5,00
Zajęcia wymagające bezpośredniego kontaktu z nauczycielem	60	2,50
Praca własna studenta (studia literaturowe, przygotowanie do zajęć laboratoryjnych/ćwiczeń, przygotowanie do kolokwium/egzaminu, wykonanie projektu)	65	2,50