



KARTA OPISU PRZEDMIOTU - SYLABUS

Nazwa przedmiotu

Analiza danych i podstawy nauczania maszynowego w języku Python [S1ETI2>ADiPNMwJP]

Przedmiot

Kierunek studiów

Edukacja techniczno-informatyczna

Rok/Semestr

3/5

Studia w zakresie (specjalność)

–

Profil studiów

ogólnoakademicki

Poziom studiów

pierwszego stopnia

Język oferowanego przedmiotu

polski

Forma studiów

stacjonarne

Wymagalność

obieralny

Liczba godzin

Wykład

15

Laboratorium

15

Inne

0

Ćwiczenia

0

Projekty/seminaria

0

Liczba punktów ECTS

2,00

Koordynatorzy

dr inż. Szymon Maćkowiak

szymon.mackowiak@put.poznan.pl

Wykładowcy

Wymagania wstępne

Podstawowa wiedza w zakresie programowania, matematyki i statystyki, umiejętność rozwiązywania prostych problemów matematycznych i technicznych w oparciu o posiadaną wiedzę, umiejętność pozyskiwania informacji ze wskazanych źródeł, zrozumienie konieczności samodzielnego dokształcania się.

Cel przedmiotu

1. Zapoznanie studentów z metodami analizowania i przetwarzania danych z wykorzystaniem języka programowania Python. 2. Przekazanie studentom podstawowej wiedzy dot. wybranych algorytmów tzw. nauczania maszynowego. 3. Zapoznanie studentów z wybranymi przykładami praktycznego zastosowania technik analizy danych, nauczania maszynowego i optymalizacji parametrycznej z wykorzystaniem języka Python.

Przedmiotowe efekty uczenia się

Wiedza:

1. Student zna aparat matematyczny niezbędny do opisu i analizy podstawowych zagadnień inżynierii materiałowej, mechaniki technicznej i informatyki, obejmujący podstawy rachunku różniczkowego i całkowego, algebrę liniową, geometrię analityczną, statystykę i metody numeryczne.

2. Student ma podstawową wiedzę w zakresie metrologii, zna i rozumie metody pomiaru wielkości fizycznych oraz analizy wyników.
3. Student ma wiedzę w zakresie programowania proceduralnego, obiektowego, sztucznej inteligencji, baz danych oraz grafiki komputerowej.

Umiejętności:

1. Student potrafi pozyskiwać informacje z literatury, baz danych oraz innych źródeł, integrować je, dokonywać ich interpretacji oraz wyciągać wnioski, formułować i uzasadniać opinie, a także przygotować i przedstawić prezentację ustną i udokumentowane opracowanie dotyczące zagadnień z zakresu kierunku kształcenia w języku polskim i angielskim.
2. Student potrafi wykorzystać nabytą wiedzę matematyczną do opisu procesów, tworzenia modeli oraz innych działań w zakresie inżynierii materiałowej, mechaniki, konstrukcji maszyn, elektrotechniki, elektroniki i informatyki.
3. Student potrafi identyfikować problem techniczny, określić jego stopień złożoności, a także zaproponować schemat jego analizy i rozwiązania.
4. Student potrafi posługiwać się obiektowymi i bazodanowymi językami programowania w zakresie aplikacji oraz konfigurowania systemów informatycznych, potrafi posługiwać się oprogramowaniem umożliwiającym graficzną prezentację i analizę wyników eksperymentalnych.
5. Student ma umiejętność samokształcenia się i rozumie potrzebę uczenia się przez całe życie zawodowe.

Kompetencje społeczne:

1. Student rozumie potrzebę ciągłego doksztalcania się (np. poprzez uczestnictwo w kursach i studiach podyplomowych) w celu podnoszenia kompetencji zawodowych i społecznych oraz potrzebę myślenia i działania w sposób przedsiębiorczy i innowacyjny.
2. Student potrafi pracować nad wyznaczonym zadaniem samodzielnie oraz współpracować w zespole przyjmując w nim różne role, wykazuje się w tej pracy profesjonalizmem i odpowiedzialnością za podejmowane decyzje.
3. Student ma świadomość roli społecznej absolwenta uczelni technicznej, a zwłaszcza rozumie potrzebę formułowania i przekazywania społeczeństwu informacji i opinii dotyczących osiągnięć technicznych i informatycznych.
4. Student postępuje zgodnie z zasadami etyki zawodowej, jest odpowiedzialny za rzetelność wyników swoich prac.

Metody weryfikacji efektów uczenia się i kryteria oceny

Efekty uczenia się przedstawione wyżej weryfikowane są w następujący sposób:

1. Projekt indywidualny.
- kryteria oceny /ocena: zgodnie z regulaminem studiów

Treści programowe

Program przedmiotu obejmuje następujące treści: korzystanie z systemów kontroli wersji GIT, korzystanie z języka Python w kontekście analizy i przetwarzania danych, korzystanie z algorytmów uczenia maszynowego do analizy wybranych typów problemów.

Tematyka zajęć

1. Konfiguracja środowiska do pracy w języku Python i system kontroli wersji GIT.
2. Podstawy języka Python.
3. Operacje matematyczne na danych oraz ich wizualizacja.
4. Operacje na danych z wykorzystaniem biblioteki Pandas.
5. Komunikacja z bazą danych SQL w języku Python.
6. Pozyskiwanie danych z Internetu.
7. Przegląd i korzystanie z wybranych zbiorów danych.
8. Przegląd wybranych algorytmów nauczania maszynowego.
9. Problem klasyfikacji.
10. Problem regresji jedno- i wielowymiarowej.
11. Problem redukcji wymiarowości zbioru danych.
12. Porównanie algorytmów nauczania nadzorowanego i nienadzorowanego.

13. Problem optymalizacji parametrów algorytmów.

14. Przegląd i porównanie wybranych bibliotek nauczania maszynowego dla języka Python.

Metody dydaktyczne

1. Prezentacja multimedialna - wprowadzenie do realizowanego tematu ćwiczeń laboratoryjnych (komputerowych).

2. Ćwiczenia laboratoryjne (komputerowe) - wykonywanie specjalistycznych programów, praca indywidualna, dyskusja.

3. Indywidualna praca projektowa studenta, dyskusja.

Literatura

Podstawowa:

1. S. Raschka, V. Mirjalili, Python. Uczenie maszynowe, wydanie 2, Helion 2019.

2. A. Saha, Matematyka w Pythonie, Helion 2015.

3. A. Sweigart, Programowanie w Pythonie dla średnio zaawansowanych, Helion 2021.

4. C. Mayer, Kod Pythona w jednym wierszu, Helion 2021.

Uzupełniająca:

1. M. Sobczyk, Statystyka, PWN 2007.

2. D. J. Rumsey, Statystyka dla bystrzaków, wyd. 2, Helion 2023.

Bilans nakładu pracy przeciętnego studenta

	Godzin	ECTS
Łączny nakład pracy	55	2,00
Zajęcia wymagające bezpośredniego kontaktu z nauczycielem	30	1,00
Praca własna studenta (studia literaturowe, przygotowanie do zajęć laboratoryjnych/ćwiczeń, przygotowanie do kolokwii/egzaminu, wykonanie projektu)	25	1,00