



**KAPITAŁ LUDZKI**  
NARODOWA STRATEGIA SPÓJNOŚCI

Projekt współfinansowany przez  
Unię Europejską w ramach  
Europejskiego Funduszu  
Społecznego

**UNIA EUROPEJSKA**  
EUROPEJSKI  
FUNDUSZ SPOŁECZNY



<b>Nazwa przedmiotu</b>		<b>Kod ECTS</b>	
Wprowadzenie do Machine Learning		11.3.1545	
<b>Nazwa jednostki prowadzącej przedmiot</b>			
Instytut Informatyki			
<b>Studia</b>			
<b>wydział</b>	<b>kierunek</b>	<b>poziom</b>	<b>pierwszego stopnia</b>
Wydział Matematyki, Fizyki i Informatyki	Informatyka	<b>forma</b>	stacjonarne
		<b>moduł</b>	wszystkie
		<b>specjalnościowy</b>	wszystkie
		<b>specjalizacja</b>	wszystkie
<b>Nazwisko osoby prowadzącej (osób prowadzących)</b>			
dr hab. Viktoriia Onyshchenko			
<b>Formy zajęć, sposób ich realizacji i przypisana im liczba godzin</b>		<b>Liczba punktów ECTS</b>	
<b>Formy zajęć</b>		5 30 godz. wykład, 30 godz. ćw. lab., praca własna studenta	
Wykład, Ćw. laboratoryjne			
<b>Sposób realizacji zajęć</b>			
zajęcia w sali dydaktycznej			
<b>Liczba godzin</b>			
Wykład: 30 godz., Ćw. laboratoryjne: 30 godz.			
<b>Termin realizacji przedmiotu</b>			
2020/2021 letni			
<b>Status przedmiotu</b>		<b>Język wykładowy</b>	
fakultatywny (do wyboru)		polski	
<b>Metody dydaktyczne</b>		<b>Forma i sposób zaliczenia oraz podstawowe kryteria oceny lub wymagania egzaminacyjne</b>	
<ul style="list-style-type: none"> <li>- Rozwiązywanie zadań</li> <li>- Wykonywanie doświadczeń</li> <li>- Wykład z prezentacją multimedialną</li> </ul>		<b>Sposób zaliczenia</b>	
		<ul style="list-style-type: none"> <li>- Zaliczenie na ocenę</li> <li>- Egzamin</li> </ul>	
		<b>Formy zaliczenia</b>	
		<ul style="list-style-type: none"> <li>- wykonanie pracy zaliczeniowej - projekt lub prezentacja</li> <li>- egzamin pisemny testowy</li> <li>- kolokwium</li> <li>- wykonanie pracy zaliczeniowej - wykonanie określonej pracy praktycznej</li> </ul>	
		<b>Podstawowe kryteria oceny</b>	
		Laboratorium:	
		<ul style="list-style-type: none"> <li>• 100% punkty z realizowanych zadań, prac domowych, projektów</li> </ul>	
		Egzamin:	
		<ul style="list-style-type: none"> <li>• 40% test wiedzy teoretycznej</li> <li>• 40% ocena realizowanych projektów</li> <li>• 20% uzyskanie certyfikatu Cisco (Python)</li> </ul>	
<b>Sposób weryfikacji założonych efektów kształcenia</b>			

zakładany efekt kształcenia		egzamin			kolokwium			projekt	
zakładany efekt kształcenia	egzamin	kolokwium	projekt	sprawdzian	referat	raport	aktywność w dyskusji	obserwacja postawy	
Wiedza									
K_W02	X	X							
P_W1	X	X						X	
Umiejętności									
K_U01			X					X	
K_U04			X					X	
P_U1			X					X	
Kompetencje społeczne									
K_K01								X	
K_Ko2							X	X	
K_K03							X	X	
P_K1							X	X	
P_K2							X	X	

sprawdzian referat raport

aktywność w dyskusji

obserwacja postawy Wiedza K\_W02 X X Umiejętności K\_U01 X X K\_U04 X X Kompetencje społeczne K\_K01 X  
K\_K02 X X K\_K03 X X

### Określenie przedmiotów wprowadzających wraz z wymogami wstępnymi

#### A. Wymagania formalne

Brak

#### B. Wymagania wstępne

Znajomość podstaw programowania, algorytmów i złożoności, języków i paradygmatów programowania obiektowego w wybranych przez studenta językach.

### Cele kształcenia

Celem kursu jest:

1. Zapoznanie z podstawowymi metodami uczenia maszynowego
2. Nabycie umiejętności stosowania podstawowych algorytmów uczenia maszynowego - praca w środowisku Python
3. Opanowanie matematycznych metod modelowania problemów i narzędzi uczenia maszynowego oraz oceny ich przydatności.

### Treści programowe

1. Wprowadzenie do podstawowej terminologii i notacji. Strategia tworzenia systemu w uczeniu maszynowym
2. Python w uczeniu maszynowym .Wykorzystywanie środowiska Python do uczenia maszynowego: Anaconda, Jupyter, NumPy, Pandas, Matplotlib, SciPy.
3. Wstępne przetwarzanie danych. Najlepsze metody oceny modelu i strojenie parametryczne.
4. Analiza danych za pomocą analizy regresyjnej, analiza skupień
5. Uczenie sieci neuronowych za pomocą biblioteki TensorFlow
6. Przykłady zastosowania :
  1. Modelowanie danych sekwencyjnych za pomocą rekurencyjnych sieci neuronowych
  2. Praca z tekstem
  3. Praca z obrazami

### Wykaz literatury

1. M. Szeliga, Data science i uczenie maszynowe, Wydawnictwo naukowe PWN, Warszawa, 2017.
2. T. Morzy, Eksploracja danych – metody i algorytmy, Wydawnictwo naukowe PWN, Warszawa, 2013.
3. Christopher M. Bishop Pattern Recognition and Machine Learning, Springer, 2007
4. J. Koronacki, J. Ćwik: Statystyczne systemy uczące się. Wydanie drugie, EXIT, Warszawa, 2007
5. K. Krawiec, J. Stefanowski, Uczenie maszynowe i sieci neuronowe. Wydawnictwo Politechniki Poznańskiej, Poznań, 2004,
6. P. Cichosz, Systemy uczące się, WNT, Warszawa, 2000.
7. W. Duch, J. Korbicz, L. Rutkowski, R. Tadeusiewicz, Sieci neuronowe, Exit, Warszawa, 2000
8. M. Gągolewski, M. Bartoszek, A. Cena, Przetwarzanie i analiza danych w języku Python, PWN, 2016

9. W. McKinney, Python for Data Analysis. Data Wrangling with Pandas, NumPy, and IPython, O'Reilly Media, 2012
10. W. Richert, L.P. Coelho, Building Machine Learning Systems with Python, Packt Publishing, 2013
11. M. Lutz, Learning Python, O'Reilly Media, 2013
12. E. Bressert, SciPy and NumPy, O'Reilly Media, 2012

<b>Kierunkowe efekty kształcenia</b>	<b>Wiedza</b>
K_W02: ma uporządkowaną, podbudowaną teoretycznie wiedzę ogólną w zakresie sztucznej inteligencji, języków formalnych, metod numerycznych	P_W1: Student zna podstawowe pojęcia w zakresie programowania, algorytmów i złożoności, języków i paradygmatów programowania (K_W02)
K_U01: potrafi zastosować wiedzę matematyczną do formułowania, analizowania i rozwiązywania problemów związanych z informatyką, projektować i analizować algorytmy pod kątem ich poprawności i złożoności obliczeniowej	<b>Umiejętności</b>
K_U04: potrafi tworzyć, uruchamiać i testować programy przy wykorzystaniu dedykowanych narzędzi oraz wzorców projektowych	P_U1: Student potrafi rozwiązywać problemy programistyczne z wykorzystaniem wiedzy matematycznej, projektować i analizować algorytmy pod kątem ich poprawności przy wykorzystaniu dedykowanych narzędzi oraz wzorców projektowych (K_U01, K_U04)
K_K01: zna ograniczenia własnej wiedzy i rozumie potrzebę dalszego uczenia się	<b>Kompetencje społeczne (postawy)</b>
K_K02: potrafi precyzyjnie formułować pytania, służące pogłębieniu własnego zrozumienia danego tematu lub odnalezieniu brakujących elementów rozumowania	P_K1: Student zdaje sobie sprawę z tempa rozwoju technologicznego i rozumie potrzebę dalszego uczenia się (K_K01, K_K02 )
K_K03: potrafi i jest gotów formułować opinie na temat podstawowych zagadnień informatycznych	P_K2: Student potrafi formułować opinie na temat zastosowania podstawowych konstrukcji programistycznych (K_K03)
<b>Kontakt</b>	
vonyshchenko@inf.ug.edu.pl	