Uniwersytet Wrocławski 30.08.2019

Metody klasyfikacji i redukcji wymiaru

Sylabus

Paweł Lorek

Opis kursu i cele

Kurs zawiera przegląd technik redukcji wymiaru (ekstrakcji cech) oraz metod klasyfikacji. Redukcja wymiaru poprawia wydajność algorytmów przetwarzania obrazów i algorytmów uczenia maszynowego, pozwala na wydajniejszą reprezentację danych i wizualizację danych wielowymiarowych. Będziemy między innymi badać analizę składowych głównych (PCA), rozkład wartości osobliwych (SVD), nieujemną faktoryzację macierzy (NMF), analizę składowych niezależnych (ICA), algorytm do wizualizacji danych t-SNE. Jeśli chodzi o metody klasyfikacji, będziemy badać wiele klasycznych klasyfikatorów, np.: metoda najbliższych sąsiadów, naiwny klasyfikator Bayesa, maszyna wektorów nośnych (SVM), liniowa i kwadratowa analiza dyskryminacyjna (LDA oraz QDA), drzewa decyzyjne, tzw. "ensemble methods" (bagging, AdaBoost, Losowe Lasy). Przedstawimy również (wraz z kilkoma zastosowaniami) model mieszanki gaussowskiej (GMM), modele ukrytych łańcuchów Markowa (HMM), modele te używają algorytmu Expectation-Maximization (EM) do nauki parametrów. Omówimy również metody oparte na Monte Carlo Markov Methods (MCMC) do " odzyskiwania " uszkodzonych obrazów. Wprowadzimy łańcuchy Markowa od podstaw, jednak podstawowa wiedza i intuicja na ich temat jest wskazana.
Podamy szczegóły dla większości wprowadzonych metod, jednakże będziemy kładli duży nacisk na intuicję i praktyczne zastosowania: omówimy (i wykorzystamy zdobytą wiedzę do różnych praktycznych problemów na laboratoriach) np. klasyfikację danych wielowymiarowych (w tym szeregów czasowych, obrazów i tekstów), kompresje obrazów, "odzyskiwanie tematów" (topic recovery), systemy rekomendujące. Kurs zapozna studentów z szerokim przekrojem algorytmów uczenia maszynowego. Studenci będą przygotowani do badań lub zastosowania nabytych metod w przemyśle.

Techniki nauki

Wykład, samodzielna praca z komputerem, projekt, prezentacja projektu, konsultacja pomysłów rozwiązywania zadań

Wymagania

Przedmioty

- Algebra
- Rachunek prawdopodobieństwa

Inne wymagania

- Podstawowa (przynajmniej) znajomość Pythona

Treści programowe

- Wprowadzenie do selekcji i ekstrakcji cech (redukcja wymiarów)
- Analiza głównych składników (PCA), rozkład wartości osobliwych (SVD), analiza składowych niezależnych (ICA). Zastosowanie do kompresji obrazu.
- Sztuczka jądra i Kernel PCA.
- Nieujemna faktoryzacja macierzy (NMF). Zastosowanie do systemów rekomendujących.
- Redukcja wymiarów przy wizualizacji danych: t-distributed Stochastic Neighbor Embedding (t-SNE), UMAP.
- Klasyczne klasyfikatory: najbliżsi sąsiedzi, naiwny klasyfikator Bayesa, mnożniki Lagrange'a - problemy pierwotne i dualne, maszyna wektorów nośnych (SVM), liniowa analiza dyskryminacyjna (LDA) zarówno jako technika redukcji wymiaru jak i klasyfikator, tworzenie klasyfikatorów wieloklasowych z dwuklasowych.
- Algorytmy grupowania. Algorytm k-means, Gaussian Mixture Model (GMM). Metoda największej wiarogodności, Algorytm Expectation-Maximization (EM). Zastosowanie do generowania obrazów (cyfr, pismo odręczne).
- Praca z tekstem, "bag of words", wektorowe reprezentacje słów (np. word2vec)
- Ukryte modele Markowa (HMM) z dyskretnymi i ciągłymi obserwacjami, procedury forward i backwart, algorytm Bauma-Welcha, algorytm Viterbiego. Zastosowania do korekcji błędów, klasyfikacji komórek w mikroskopii fluorescencyjnej oraz do klasyfikacji szeregów czasowych.
- Wprowadzenie do metod Monte Carlo Markov Chain (MCMC). Próbnik Gibbsa. Zastosowanie do "odzyskiwania" uszkodzonych (w określony, znany sposób) obrazów.

Zakładane efekty kształcenia

Wiedza

- Student zna szereg technik redukcji wymiaru
- Student zna podstawowe algorytmu klasyfikacji "płytkiego-uczenia się"
- Student zna podstawowe algorytmy klasteryzujące
- Student umie analizować złożoność zadanych problemów oraz jak dokonywać stosownych optymalizacji

Umiejętności

- Student potrafi zaprojektować i wdrożyć rozwiązania oparte o algorytmy uczenia maszynowego
- Student jest zaznajomiony z zastosowaniami algorytmów uczenia maszynowego do rzeczywistych problemów, np. systemów rekomendacjących, klasyfikacji tekstu, kompresji obrazu, wizualizacji danych
- Student potrafi uporządkować i przygotować dokumentację naukową i techniczną opisującą działania podjęte podczas wykonywania projektu
- Student potrafi samodzielnie poszerzyć wiedzę zdobytą podczas kursu poprzez czytanie i rozumienie dodatkowego materiału (dokumentacji naukowej i technicznej)

Metody weryfikacji zakładanych efektów kształcenia

dwa projekty, implementacja i prezentacja programów komputerowych, ustny egzamin (prezentacja i odpowiadanie na pytania), pisanie raportów

Warunki i forma zaliczenia

- *Laboratoria*: Będą 2 projekty. Aby zdać trzeba uzyskać minimalną liczbę punktów. Oddanie projektu = oddanie raportu (w .pdf) oraz działającego programu.
- *Ustny egzamin*: Przedstawienie własnych projektów i odpowiadanie na pytania.

Nakład pracy studenta

- Zajęcia z nauczycielem:
  - wykłady - 30 godzin
  - laboratoria - 30 godzin

- Praca własna studenta:
  - przygotowywanie się do zajęć: - 15 godzin
  - czytanie dodatkowego materiału - 30 godzin
  - implementacja algorytmów, pisanie raportów - 45 godzin

Literatura

[1] Andreas C. Muller; Sarah Guido *Introduction to machine learning with Python: a guide for data scientists*, Sebastopol O'Reilly Media, Inc., 2017

[2] Matthew Kirk *Thoughtful Machine Learning With Python*, Sebastopol O'Reilly Media, Inc., 2017

[3] Olle Häggström. *Finite Markov Chains and Algorithmic Applications*, Cambridge University Press, 2002. (Rozdziały 1–7)