Metody arbitrażu statystycznego

Michał Sierakowski

IBM, University of Warsaw

m.sierakowski@mimuw.edu.pl
Slack channel: https://ibm.enterprise.slack.com/archives/C09KE8UFH33

4 listopada 2025r.

Przegląd

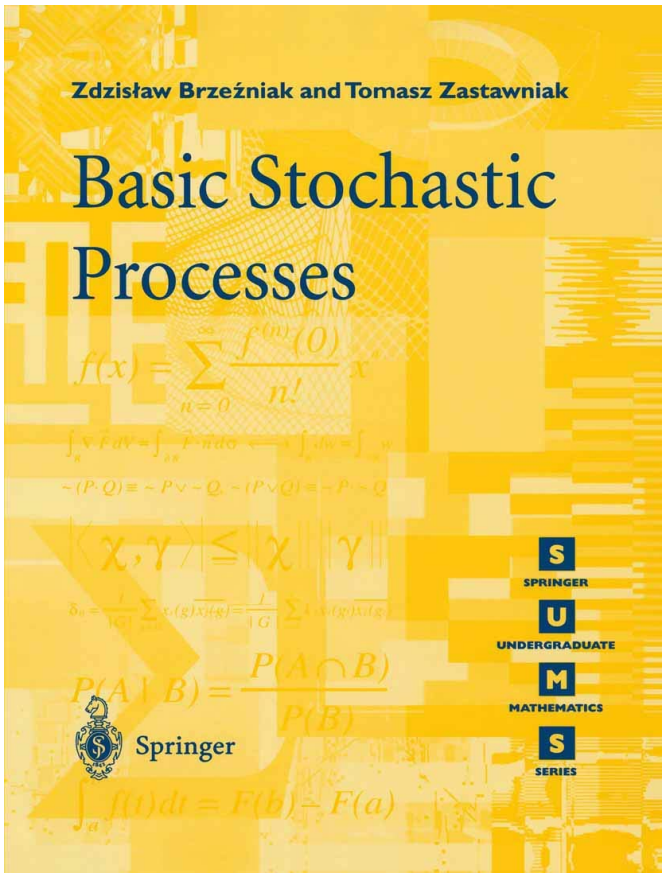
Zasady

- **ocena końcowa** = 40% ćwiczenia + 10% kolokwium + 15% prezentacja + 35% projekt
  - $[0 - 50\%)$: ocena 2
  - $[50\% - 60\%)$: ocena 3
  - $[60\% - 70\%)$: ocena 3,5
  - $[70\% - 80\%)$: ocena 4
  - $[80\% - 90\%)$: ocena 4,5
  - $[90\% - 100\%]$: ocena 5

Zasady: ćwiczenia (waga 40%)

- **indywidualne zadania w trakcie zajęć**
  - punktowane: $\{0, 1\}$
  - zrzut ekranu i wyjaśnienie na zajęciach
- **grupowe zadania w trakcie zajęć**
  - punktowane: $\{0, 1\}$ tylko dla obecnych na zajęciach członków grupy (co należy zaznaczyć zgłaszając rozwiązanie)
  - zrzut ekranu i wyjaśnienie na zajęciach
- **indywidualne zadania domowe**
  - punktowane: $\{0, 1\}$
  - czas wykonania: do rozpoczęcia kolejnych zajęć
  - zrzut ekranu lub listing
- **grupowe zadania domowe**
  - punktowane: $\{0, 1\}$ dla wszystkich członków grupy
  - czas wykonania: do rozpoczęcia kolejnych zajęć
  - zrzut ekranu lub listing

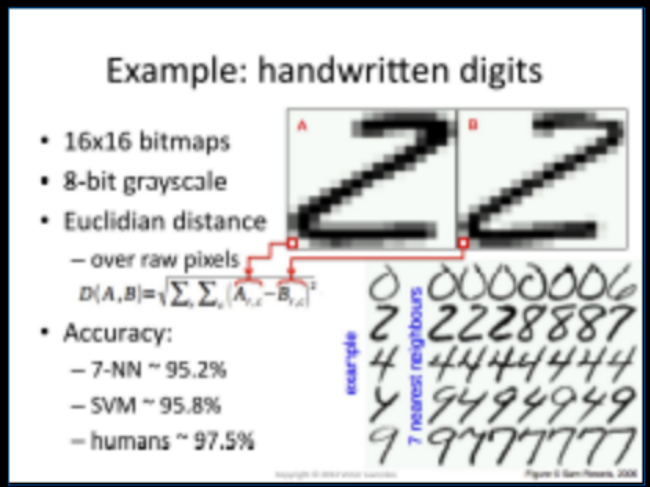
Motywacja

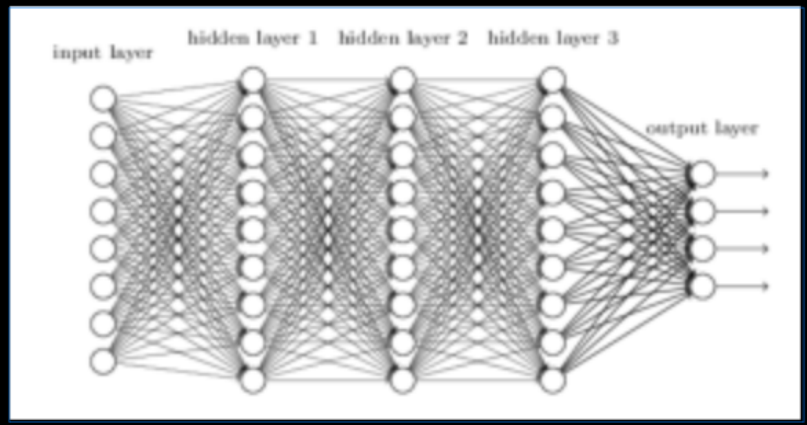

Zdzisław Brzeźniak and Tomasz Zastawniak
Basic Stochastic Processes
SPRINGER
UNDERGRADUATE
MATHEMATICS
SERIES
Springer


L'IGNORANZA
STECCALEGNE
ONORANZA
P I Z Q
T Z Q
TVTTI
Wstęp
do
teorii
prawdopodobieństwa
Wydanie II
JACEK JAKUBOWSKI
RAFAŁ SZTENCEL
SCRIPT

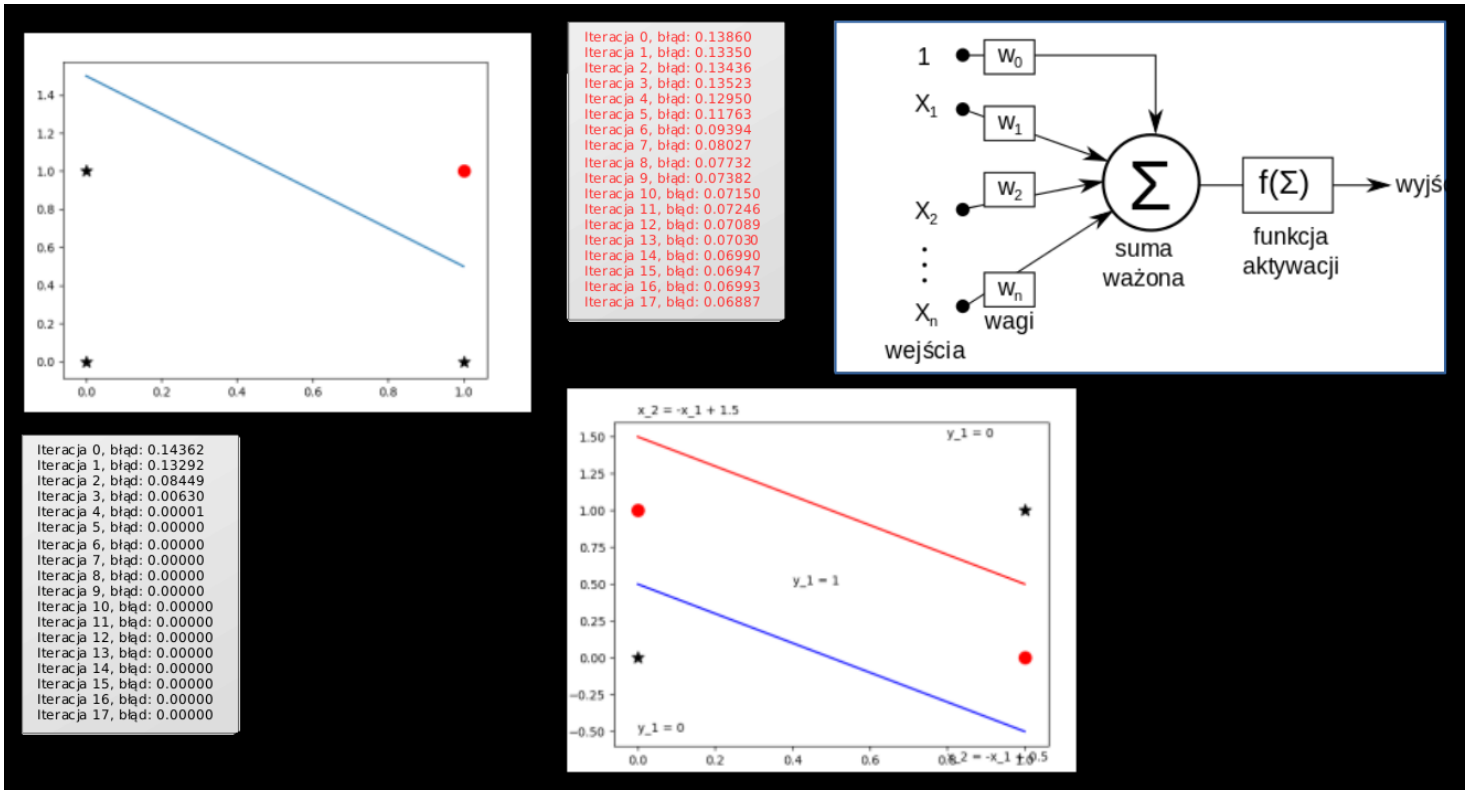
Pragmatycznie o AI

- ✔ **Machine Learning**
  - ✔ Supervised ML
  - ✔ Unsupervised ML
- ✔ **Deep Learning**
- ✔ Reinforcment Learning

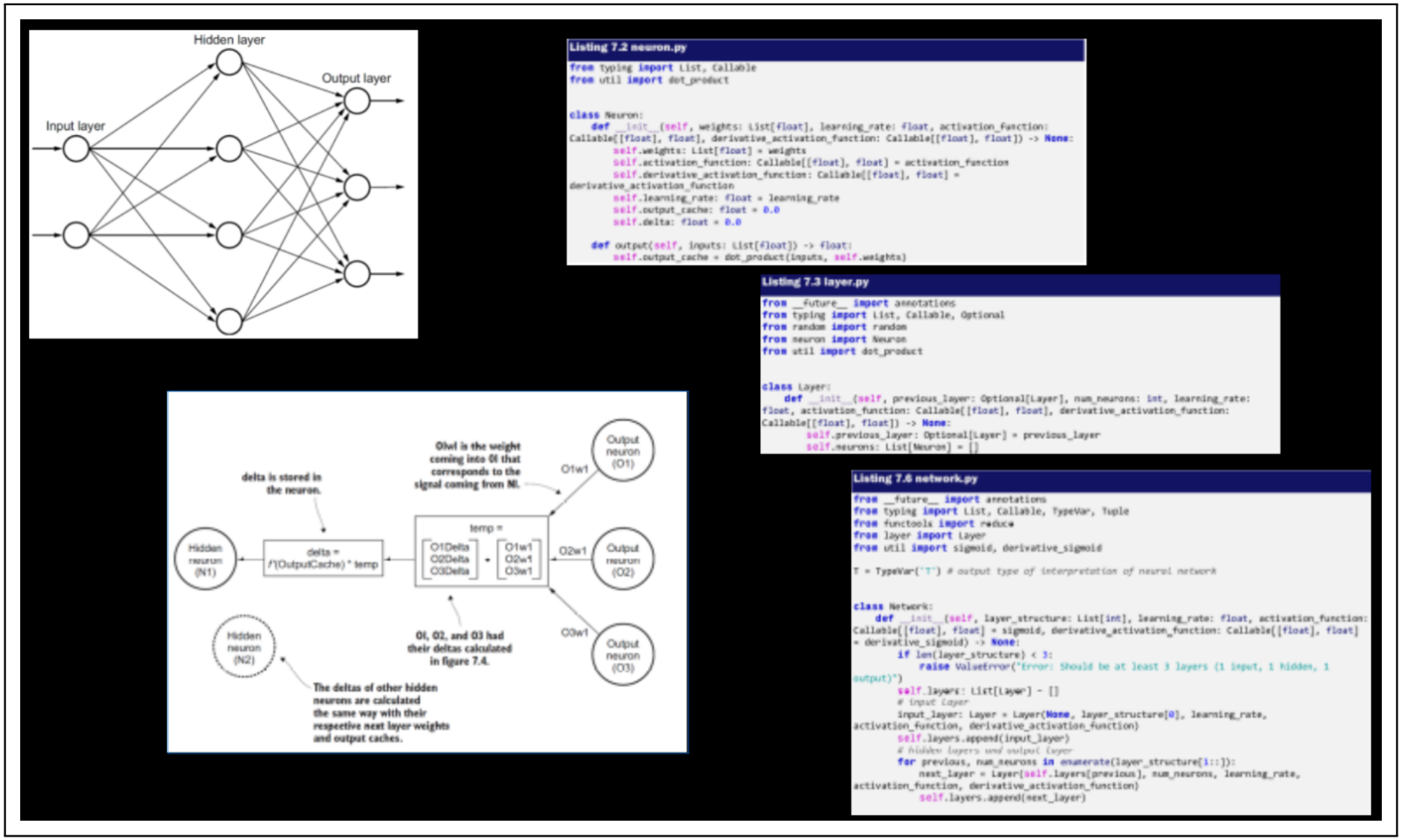




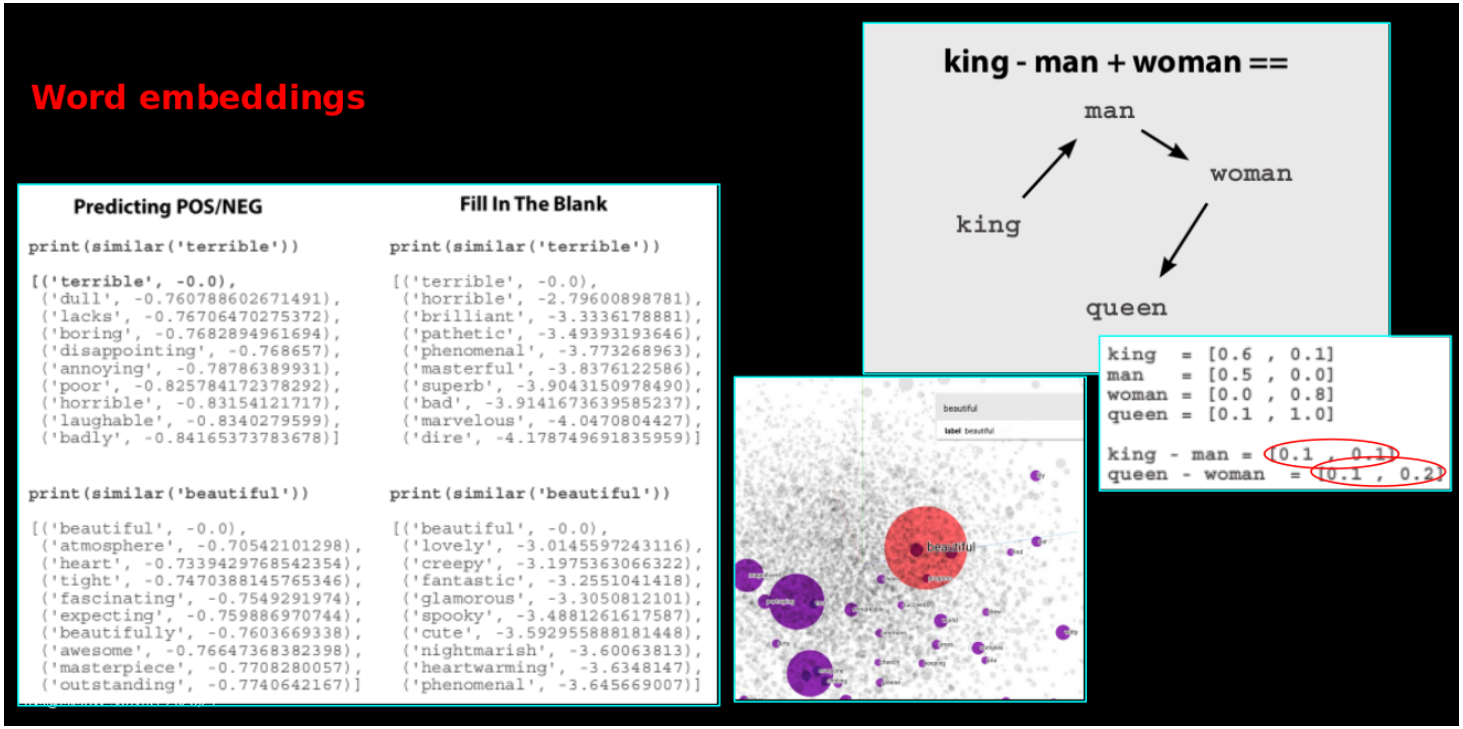
Pragmatycznie o AI



Pragmatycznie o ML



Osadzanie słów



Rysunek: Osadzanie słów

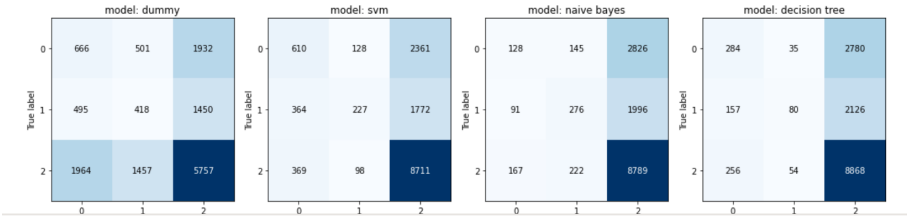
Współczesne modele NLP

	Word order awareness	Context awareness (cross-words interactions)
Bag-of-unigrams	No	No
Bag-of-bigrams	Very limited	No
RNN	Yes	No
Self-attention	No	Yes
Transformer	Yes	Yes

Rysunek: Modele NLP

Klasyfikacja wieloklasowa

```
model=dummy, f1_micro: mean=0.47, std=0.01
model=svm, f1_micro: mean=0.65, std=0.0
model=naive bayes, f1_micro: mean=0.63, std=0.0
model=decision tree, f1_micro: mean=0.63, std=0.0
model=random forest, f1_micro: mean=0.63, std=0.0
```

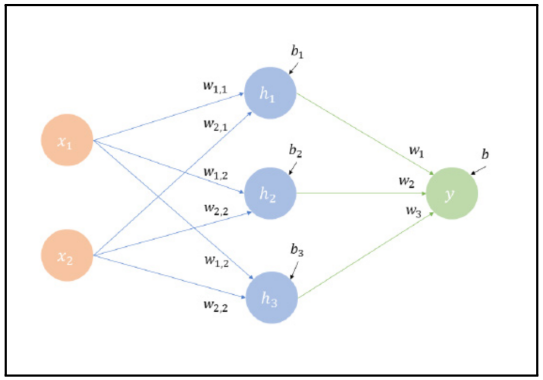


Lab I.50 klasyfikacja wieloklasowa

Wykonaj polecenia zawarte w dołączonym notatniku `Lab.I.50.raw.ipynb`.

Uwaga: Przejrzyj również plik `helper.py` w celu poprawienia wyników zawartych w przykładach z powyższego notatnika.

Prosta sieć neuronowa



Lab I.51 Prosta sieć neuronowa

Po prostu wykonaj polecenia w notatniku `Lab.I.51.raw.ipynb` i pobierz utworzony w pakiecie Keras model sieci neuronowej. Wykorzystaj bibliotekę `ibm_watson_studio_lib`.

Zamiast Lab I.32

Lab H.I.52 Proszę poprawnie wykonać wszystkie w kroki w dołączonym notatniku `Lab.H.I.52.raw.ipynb` wykorzystując inny niż w kodzie model LLM dostępny w IBM Cloud.

The End