Predicting the class of Haberman's survival with neural networks

Adrian Moczulski, Rafał Kitta

Plan prezentacji

1. Opis projektu
2. Aktualny stan prac
3. API reference
4. Sposoby walidacji sieci neuronowych

Opis projektu

- Celem project jest wytrenowanie sieci neuronowej tak aby przewidywała czy pacjent przeżyje po operacji raka piersi

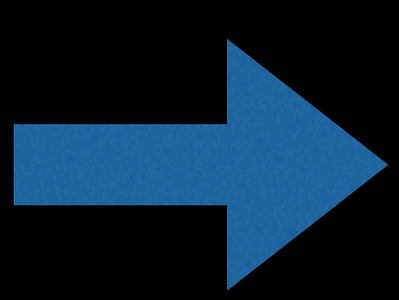
- Problem klasyfikacji

Zbiór danych treningowych

- Haberman's Survival Dataset (March 4, 1991)
- Dane pochodzą z University of Chicago's Billings Hospital z lat 1958-1970
- Zbiór posiada 306 instancji
- Dane wejściowe:
  - wiek pacjenta
  - rok operacji
  - ilość wykrytych pozytywnych węzłów pachowych
- Dane wyjściowe - przeżywalność po 5 latach po operacji

Zbiór danych treningowych

Wiek	Rok operacji	Węzły	1=przeżył
30	64	1	1
30	62	3	1
30	65	0	1
31	59	2	1
31	65	4	1
33	58	10	1
33	60	0	1
34	59	0	2
34	66	9	2
34	58	30	1
34	60	1	1
34	61	10	1
34	67	7	1



Wiek	Rok operacji	Węzły	1=przeżył
0	0,54	0,01	1
0	0,36	0,05	1
0	0,63	0	1
0,0	0,09	0,03	1
0,0	0,63	0,07	1
0,0	0	0,19	1
0,0	0,18	0	1
0,0	0,09	0	0
0,0	0,72	0,17	0
0,0	0	0,57	1
0,0	0,18	0,01	1
0,0	0,27	0,19	1
0,0	0,81	0,13	1

Normalizacja

Klasyczny przykład Min Max

B=((A-min value of A)/(max value of A-min value of A))*(D-C)+C

- B – wartość znormalizowana
- A – wartość normalizowana
- D i C – determinują zakres wartości wyjściowych

w naszym przypadku: D= 0 and C=1

Aktualny stan prac

- Implementacja sieci neuronowej w języku Swift
- Aplikacja konsolowa na system macOS
- Wczytywanie znormalizowanych danych z pliku .csv, mapowanie ich do obiektów
- Walidacja *Leave-one-out cross-validation*
- W pełni udokumentowane API

API reference

- Tworzone jest dynamicznie z komentarzy w kodzie
- Prezentowane w formie estetycznej i minimalistycznej strony www
- Uwzględnia klasy widoczności
- Generowane za pomocą narzędzia Jazzy ♪♫

Demo

Sposoby walidacji sieci neuronowych

$$\text{jakość uczenia} := \frac{\text{ilość przykładów sklasyfikowanych poprawnie}}{\text{ilość wszystkich przykładów}}$$

Sposoby walidacji sieci neuronowych

- dane uczące są losowo dzielone na dwa rozłączne zbiory:
  - próbkę uczącą U,
  - próbkę testową T,
- sieć jest uczona za pomocą próbki uczącej,
- jakość sieci jest badana tylko za pomocą próbki testowej

$$\text{jakość} := \frac{\text{ilość przykładów } T \text{ sklasyfikowanych poprawnie}}{\text{ilość wszystkich przykładów w } T}$$

Cross-Validation

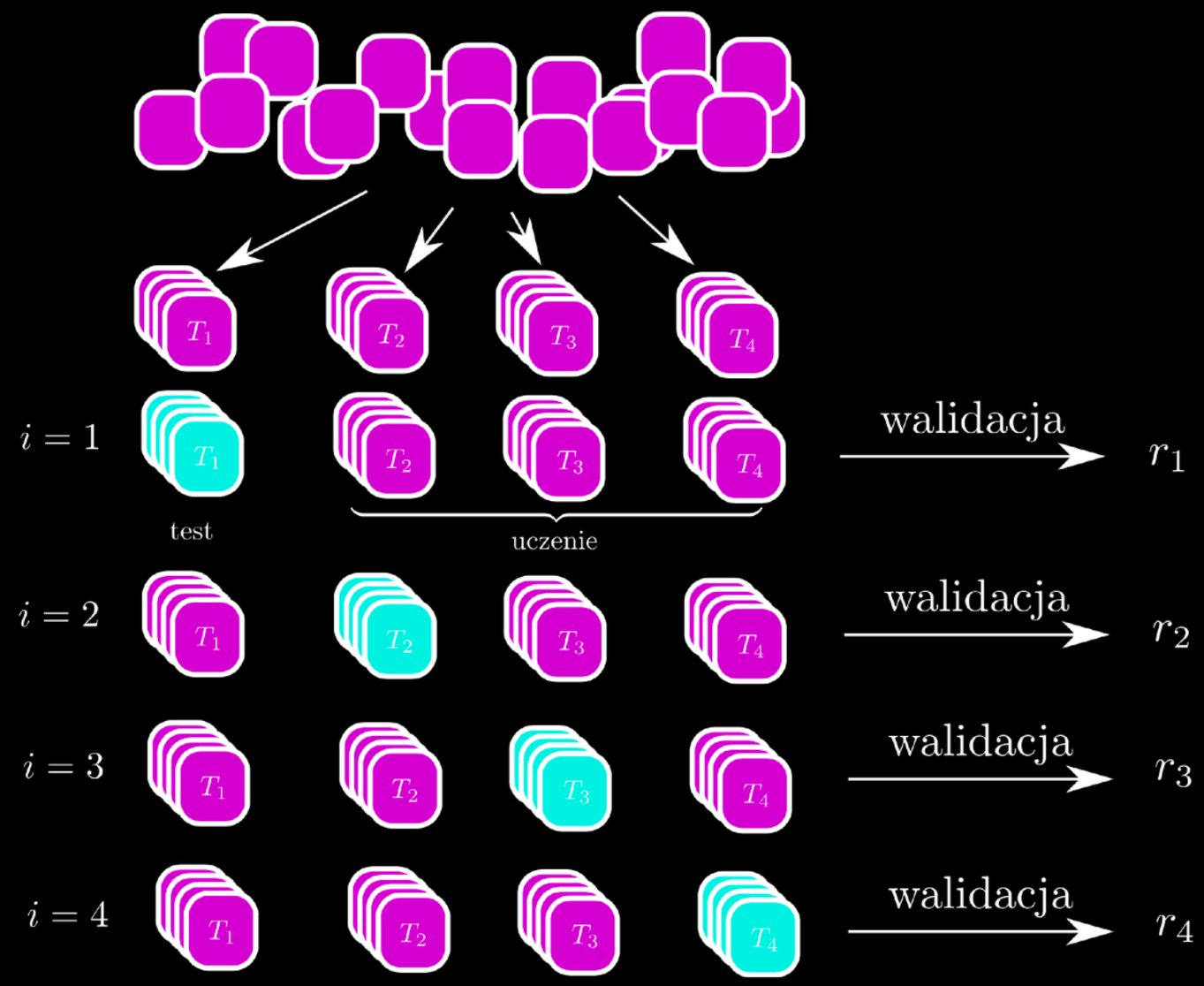
- Exhaustive cross-validation (wyczerpująca)
  - Leave-p-out cross-validation
  - Leave-one-out cross-validation
- Non-exhaustive cross-validation (niewyczerpująca)
  - k-fold cross-validation
  - 2-fold cross-validation
  - Repeated random sub-sampling validation

Sposoby walidacji sieci neuronowych

Ang. *k-fold cross-validation*

- dane uczące są losowo dzielone na k rozłącznych zbiorów: $T_1, ..., T_k$,
- zbiory powinny być równoliczne (lub różnić się o maksymalnie 1 element, jeżeli nie da się podzielić dokładnie),
- dla $i = 1...k$ powtarzamy
  - uczymy sieć na zbiorze uczącym $T_1 \cup ... T_{i-1} \cup T_{i+1} \cup T_k$,
  - testujemy tak nauczoną sieć na danych T_i (na tych danych sieć nie była uczona),
  - zapamiętujemy rezultat jako r_i
- zależnie od ilości miejsca podajemy wszystkie rezultaty r_i,
- lub **przynajmniej** ich średnią, medianę, minimum, maksimum i odchylenie standardowe,

Sposoby walidacji sieci neuronowych



http://www-users.mat.uni.torun.pl/~piersaj/www/contents/teaching/wsn2010/wsn-lec13-walidacja.pdf

Leave one out

- odmiana walidacji krzyżowej, w której $k =$ ilość elementów w T,
- dla $i = 1...n$ powtarzamy:
  - uczymy sieć na zbiorze uczącym $T \setminus T_i$,
  - testujemy sieć na pozostałym przykładzie T_i,
  - zapamiętujemy wynik r_i (będzie on albo $+1$, albo 0),
- obliczamy średnią i odchylenie standardowe wyników,
- można stosować w przypadku małej ilości danych w zbiorze T.

Leave one out

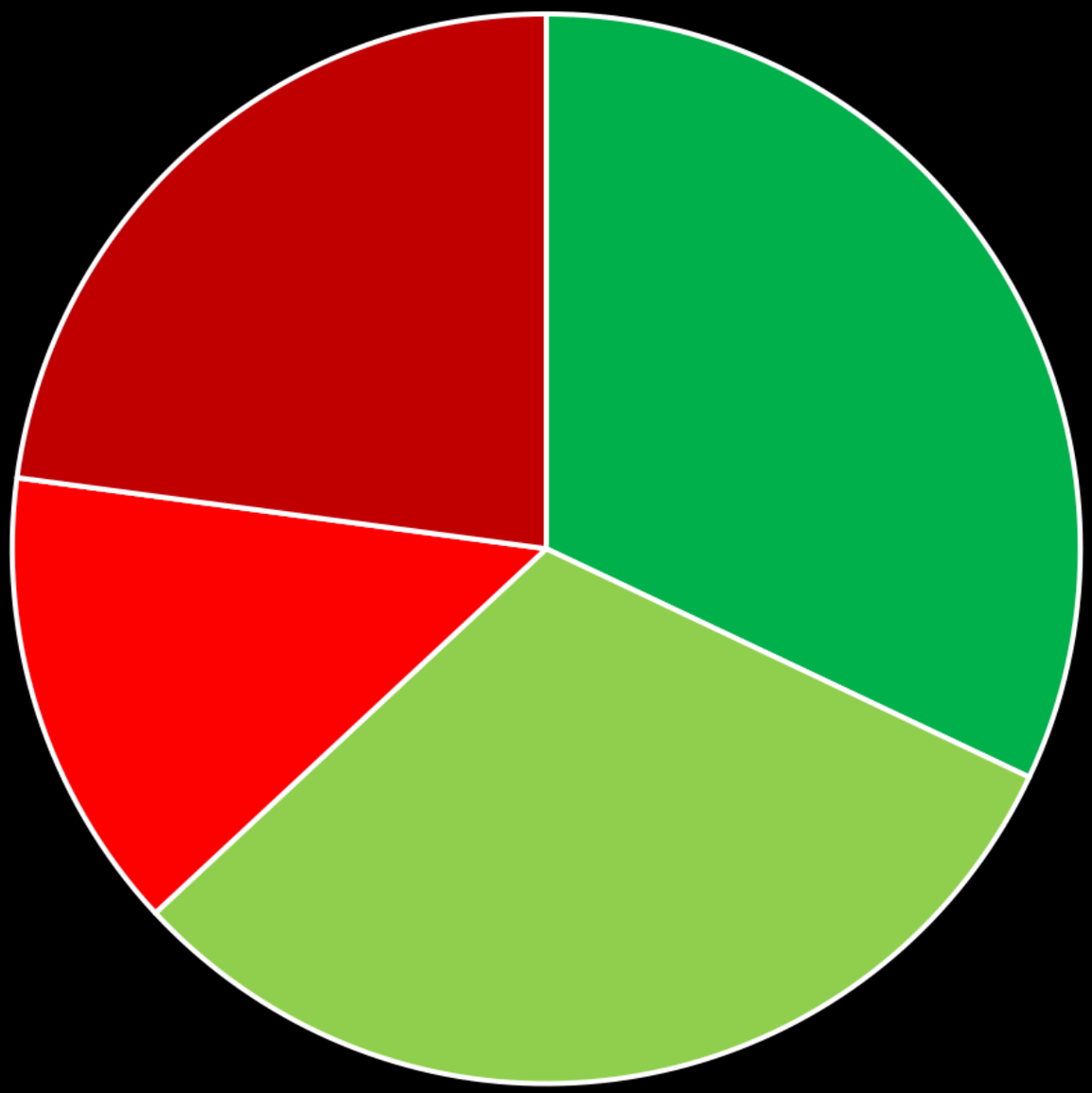
Odchylenie od wartości docelowej	Klasyfikacja zdania testu
0,229912906	1
0,065565117	0
0,231073266	1
0,040247537	0
0,355065264	1
0,006595685	0
0,394426375	1
-0,824843619	3
-0,814627405	3
0,037859567	0
0,0430196	0
0,517428345	2
0,009244744	0

Odchylenie	Klasyfikacja
< 0.2	0
(0.2; 0.5>	1
(0.5; 0.8>	2
> 0.8	3

Statystyki

Liczba testów definitywnie zdanych	98
Liczba testów prawie zdanych	95
Liczba testów delikatnie niezdanych	43
Liczba testów definitywnie niezdanych	70

Statystyki


Liczba testów definitywnie zdanych
Liczba testów prawie zdanych
Liczba testów delikatnie niezdanych
Liczba testów definitywnie niezdanych