

Głębokie uczenie w praktyce	Data: 16.06.2025
Grupa PS2 Patryk Wójtowicz	Prowadzący: dr hab. inż. prof. PB Jacek Grekow

Dokumentacja projektu: DeathRiskAI

1. Opis danych uczących i wstępnej obórki danych

Zbiór danych wykorzystywany w projekcie pochodzi z pliku `gold.csv` i zawiera 44 000 rekordów, reprezentujących dane medyczne pacjentów. Klasa docelowa `death` jest binarna i opisuje ryzyko zgonu:

- `0` oznacza brak ryzyka,
- `1` oznacza wysokie ryzyko zgonu.

Rozkład klas był silnie niezbalansowany:

- Klasa `0`: ~6 500 rekordów,
- Klasa `1`: ~3 600 rekordów.

Wstępna obórka:

- Dane klasy `0` zostały oczyszczone z wartości odstających przy użyciu z-score (> 3.0).
- Klasa `1` została pozostawiona bez usuwania outlierów ze względu na jej niewielką liczebność.
- Następnie zastosowano skalowanie `MinMaxScaler`, a dane podzielono na zbiory:
 - `trainval` (90%)
 - `test` (10%)

SMOTE:

Na zbiorze treningowym zastosowano technikę oversamplingu SMOTE w celu zbalansowania klas, co umożliwiło skuteczniejsze trenowanie modelu.

2. Opis konstrukcji sieci neuronowej

Model oparty jest na sieci neuronowej zbudowanej w TensorFlow i Keras. Przewidziano dwa warianty:

a) Model bazowy:

```
Input -> Dense(128, relu) -> Dropout(0.3)
      -> Dense(128, relu) -> Dropout(0.3)
      -> Dense(64, relu) -> Dropout(0.2)
      -> Dense(16, relu) -> Dropout(0.2)
      -> Dense(1, sigmoid)
```

b) Model strojony (KerasTuner):

- Liczba warstw: od 2 do 5,
- Liczba neuronów: 32 do 256,
- Funkcje aktywacji: `relu`,
- Dropout: 0.0 do 0.5,
- Learning rate: 1e-4 do 1e-2.

Zapis konfiguracji najlepszego modelu dokonywany jest automatycznie do pliku `best_hp.json`.

3. Wpływ wariantów trenowania i budowy sieci na wyniki ewaluacji

a) Próg decyzyjny:

Zastosowano automatyczny dobór progu klasyfikacji na podstawie F1-score (0.0–1.0, krok co 0.01). Próg ten zapewnia maksymalizację trafności detekcji klasy 1.

b) SMOTE + klasy wagowe:

Zastosowanie SMOTE poprawiło recall i f1-score dla klasy 1 bez konieczności stosowania wysokich `class_weight`. W testach SMOTE był skuteczniejszy i bardziej stabilny niż samo użycie `class_weight`.

c) Strojenie hiperparametrów:

KerasTuner pozwolił uzyskać istotną poprawę F1-score (np. z ~0.83 do ~0.93) dla klasy 1. Najlepsze konfiguracje charakteryzowały się:

- 3 warstwami,
- 128 neuronami na warstwę,
- aktywacją `relu`,
- dropoutem 0.2–0.3,
- learning rate ~0.001.

4. Ewaluacja

Metryki obliczane są dla zbioru testowego przy użyciu optymalnego progu klasyfikacji:

- **Accuracy**,
- **Precision**,
- **Recall**,
- **F1-score**,

Raporty zapisywane są w formatach:

- **CSV** (`test_predictions.csv`, `test_metrics.csv`),
- **PNG** (wykres metryk),
- **TXT** (czytelna tabela metryk).

Przykładowe wyniki:

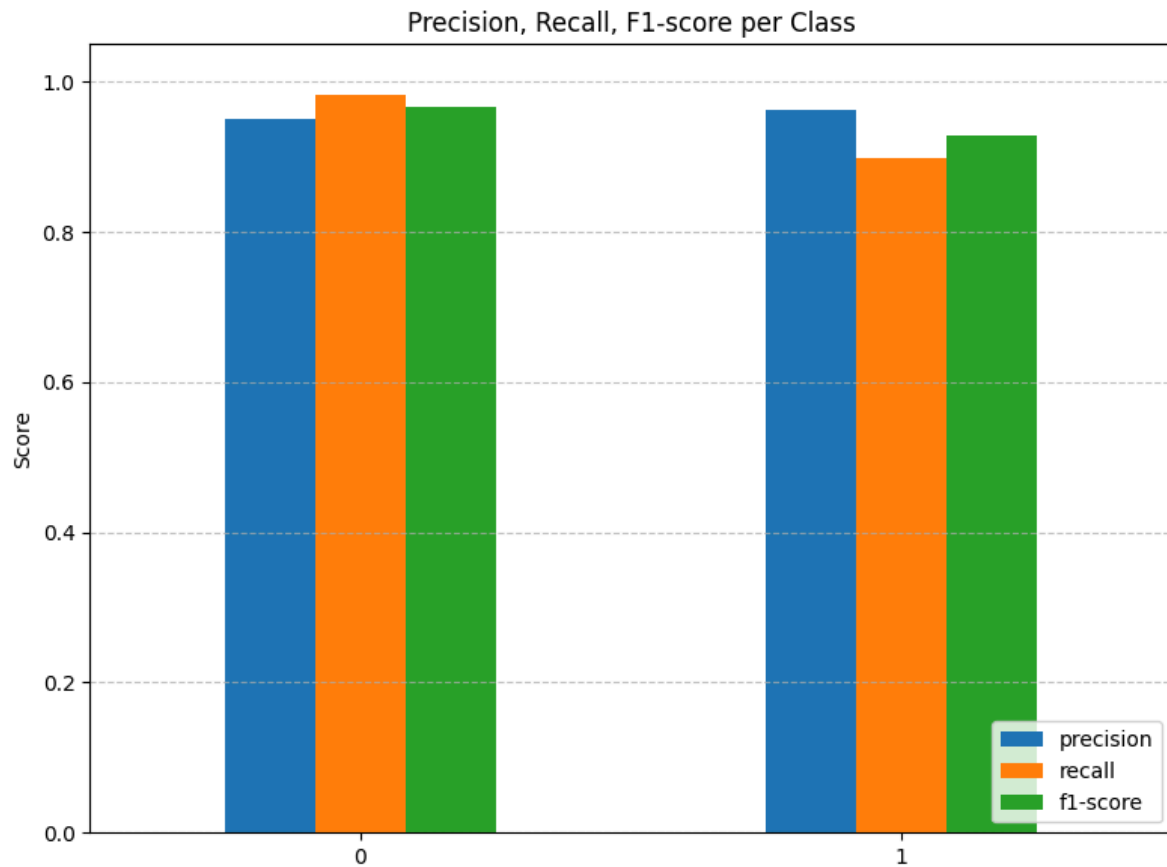
Najlepsze hiperparametry:

```
{
  "num_layers": 5,
  "units_0": 224,
  "activation": "relu",
  "dropout_0": 0.30000000000000004,
  "units_1": 256,
  "dropout_1": 0.30000000000000004,
  "lr": 0.0004986787447507789,
  "units_2": 256,
  "dropout_2": 0.1,
  "units_3": 64,
  "dropout_3": 0.0,
  "units_4": 128,
  "dropout_4": 0.4,
  "tuner/epochs": 20,
  "tuner/initial_epoch": 7,
  "tuner/bracket": 2,
  "tuner/round": 2,
  "tuner/trial_id": "0012"
}
```

Dobrany threshold:

```
{"best_threshold": 0.72, "f1_score": 0.9288702928870293}
```

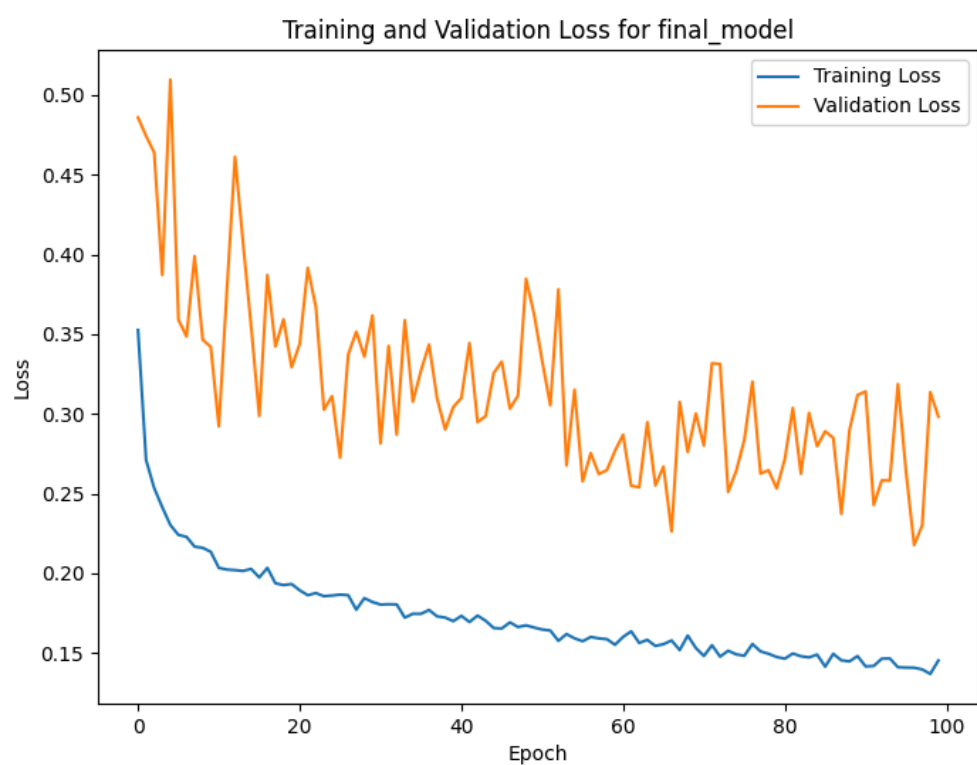
Metryki wykres:



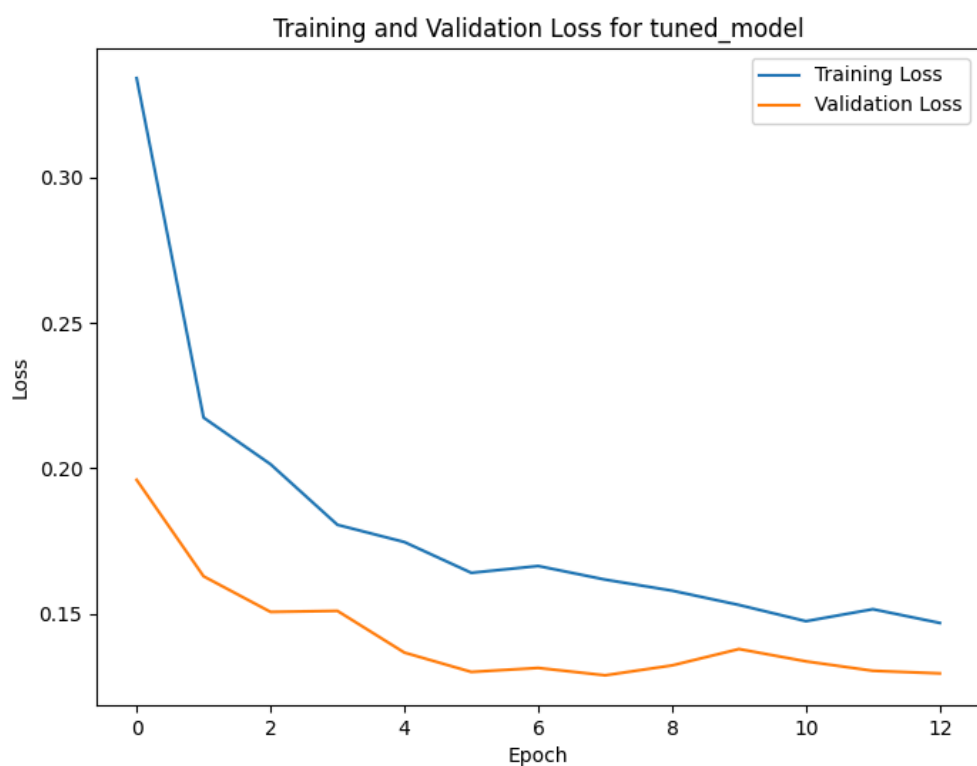
 Classification Report on Test Data

Class	Precision	Recall	F1-score	Support
0	0.95	0.98	0.97	731
1	0.96	0.90	0.93	371
accuracy		0.95		1102
macro avg	0.96	0.94	0.95	1102
weighted avg	0.95	0.95	0.95	1102

Wykres strat treningu modelu:



Wykres strat tuningu modelu:



Struktura modelu:

