AutoML

Auto-PyTorch

Marcel Witas
Łukasz Tomaszewski
Adam Frej

28 maja 2022

Plan prezentacji

1. AutoML
2. Auto-PyTorch
3. Własny pipeline
4. Podsumowanie

Plan prezentacji

1. AutoML
2. Auto-PyTorch
3. Własny pipeline
4. Podsumowanie

AutoML na danych tabelarycznych

- Automatyczny framework do uczenia maszynowego
- Dane w formie tabelarycznej
- Cel: realizacja pełnego uczenia maszynowego od preprocessingu do trenowania modeli i znajdowania jak najlepszej predykcji bez żadnej ingerencji użytkownika.
- Rozpowszechnianie uczenia maszynowego wśród nietechnicznych osób.

AutoML - wymagania

- Minimalny preprocessing potrzebny tylko do uruchomienia uczenia.
- Brak strojenia hiperparametrów - dostępne wyłącznie techniczne parametry (np. czas uczenia, liczba wątków).
- Zwracanie gotowego modelu i obliczonej predykcji.
- Dostęp do kroswalidacji i możliwość definiowania własnego podziału na foldy.

Plan prezentacji

1. AutoML
2. **Auto-PyTorch**
3. Własny pipeline
4. Podsumowanie

Auto-PyTorch

- "Auto Deep Learning"
- Połączenie tradycyjnych modeli ML z sieciami neuronowymi
- Skupiony na dobieraniu i strojeniu sieci

Auto-PyTorch - etapy

- Prosty preprocessing - obsługa danych liczbowych i kategorycznych
- Trenowanie tradycyjnych modeli przy użyciu Multi-Fidelity Bayesian Optimization
- Poszukiwanie odpowiedniej architektury sieci neuronowej i strojenie jej hiperparametrów
- Ważony ensembling wszystkich modeli

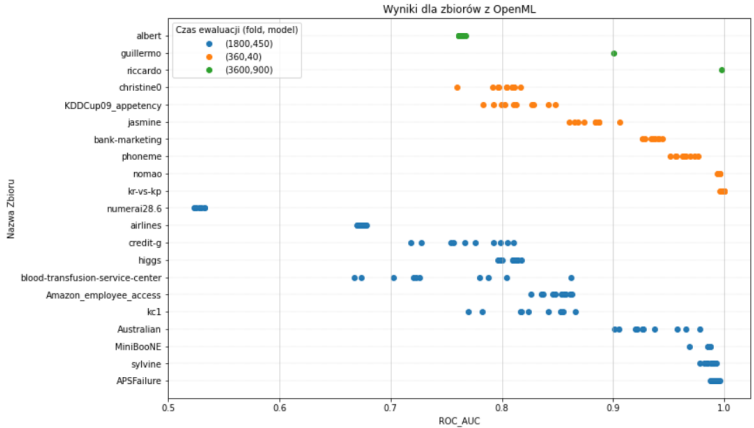
Auto-PyTorch - cechy charakterystyczne

- Portfolia - predefiniowane zestawy konfiguracji - warmstart optymalizacji
- Połączenie optymalizacji bayesowskiej i hyperband'u
- Obsługa wielowątkowości

Auto-PyTorch - odczucia

- Intuicyjny w użyciu
- Minimalne przygotowanie danych
- Problem z parametrem dotyczącym limitu czasu

Auto-PyTorch - wyniki na zbiorach OpenML



ROC AUC AutoPyTorcha dla zbiorów dotyczących klasyfikacji binarnej z OpenML

Plan prezentacji

1. AutoML
2. Auto-PyTorch
3. **Własny pipeline**
4. Podsumowanie

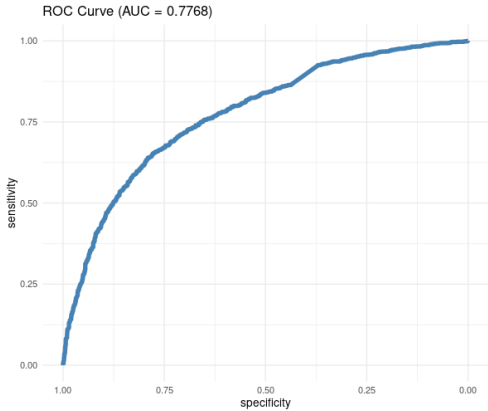
Opis pipeline'u

- Problem: klasyfikacja binarna
- Preprocessing:
  - Usunięcie duplikatów
  - Imputacja brakujących zmiennych
  - Encoding zmiennych kategorycznych
  - Usunięcie zmiennych o niskiej wariancji
  - Skalowanie zmiennych do przedziału [0,1]

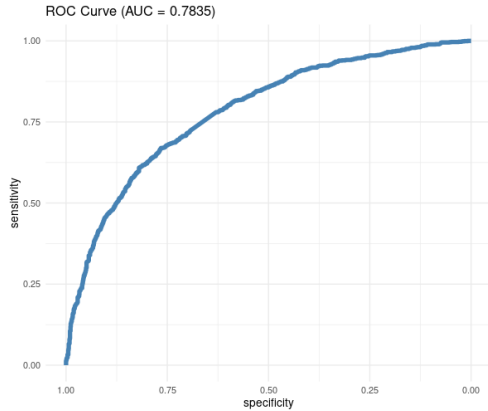
Opis pipeline'u cd.

- Przygotowanie modeli:
  - Użycie SVC, XGBoost oraz RandomForestClassifier
  - Wyznaczanie najlepszych hiperparametrów przy użyciu Optymalizacji Bayesowskiej
  - Wykorzystanie modeli deep learningowych - Keras
  - Ensembling modeli - Soft Voting
- Trenowanie otrzymanego modelu

Wyniki dla benchmarku z zajęć



Krzywa ROC dla naszego pipeline'u
(czas ewaluacji - 1 godzina)



Krzywa ROC dla Auto-PyTorch
(czas ewaluacji - 10 minut)

Plan prezentacji

1. AutoML
2. Auto-PyTorch
3. Własny pipeline
4. **Podsumowanie**

Podsumowanie

- Celem AutoML jest realizacja pełnego uczenia maszynowego i znalezienie jak najlepszej predykcji bez ingerencji użytkownika.
- Auto-PyTorch łączy tradycyjne modele ML z sieciami neuronowymi i wykonuje ensembling znalezionych modeli. Wykorzystuje portfolia oraz łączy optymalizację bayesowską z hyperbandem.
- Auto-PyTorch osiągnął bardzo dobre wyniki na zbiorach z OpenML, jednak preblematycznym okazało się ustawienie odpowiedniego czasu ewaluacji.
- Stworzony przez nas pipeline wykonuje podstawowy preprocessing, trenuje modele i wykonuje ensembling.
- Dla testowanego zbioru nasz pipeline osiągnął wyniki podobne do Auto-PyTorcha, jednak ewaluacja trwała o wiele dłużej.

Dziękujemy za uwagę.