AutoML: Praca domowa 1

Jan Cichomski 313201
Adam Dąbkowski 313212

1 Wstęp

W sprawozdaniu skupimy się na analizie tunowalności hiperparametrów trzech wybranych algorytmów uczenia maszynowego: Decision Tree, ElasticNet i Random Forest. Głównym celem jest zbadanie, jak skutecznie można dostosować parametry tych modeli, korzystając z siedmiu różnych zbiorów danych 5.4. Dodatkowo, eksplorujemy tunowanie modeli przy użyciu techniki optymalizacji bayesowskiej (Bayes optimalization) oraz losowej metodzie wybierania punktów (Random Search).

2 Opis eksperymentu

Dla każdego modelu przeprowadzono następujący eksperyment:

1. Wylosuj $N = 50$ z przestrzeni hiperparametrów danego modelu
2. Znajdź hiperparametrów, które dają średnio najlepsze wyniki na wielu zbiorach danych (domyślne hiperparametry)
3. Dla każdego zbioru danych znajdź najlepszą konfigurację hiperparametrów przy użyciu Random Search na wcześniej wylosowanych $N = 50$ punktach
4. Dla każdego zbioru danych znajdź najlepszą konfigurację hiperparametrów przy użyciu optymalizacji Bayesowskiej z wykorzystaniem $N = 50$ kroków
5. Oblicz tunowalność modelu dla każdego zbioru danych, na podstawie domyślnych parametrów i znalezionej najlepszej konfiguracji na danym zbiorze danych znalezionej przez Random Search
6. Oblicz tunowalność modelu dla każdego zbioru danych, na podstawie domyślnych parametrów i znalezionej najlepszej konfiguracji na danym zbiorze danych znalezionej przez optymalizacji Bayes'a

Przyjęta miara jakości modelu to R^2 (Coefficient of determination).

2.1 Przestrzeń hiperparametrów

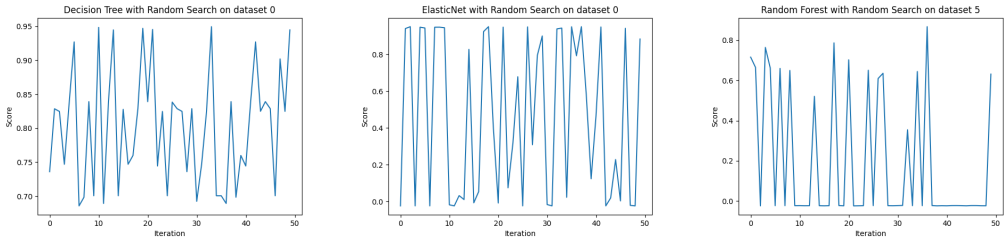
Przestrzeń hiperparametrów, w której optymalizowane poszczególne modele zaczerpnięto z pracy „Tunability: Importance of Hyperparameters of Machine". Można je również znaleźć w dodatku 5.1.

2.2 Stabilność algorytmów samplingu

2.2.1 Sampling z Random Search

Sampling punktów z przestrzeni hiperparametrów przy użyciu Random Search, tak jak można było przypuszczać, jest bardzo losowy. Z reguły już po sprawdzeniu kilku punktów (ok. 5), metoda daje satysfakcjonujące wyniki. Dalsze iteracji mają

tendencję poprawić jakoś hiperparametrów, ale jest to prawie niezauważalna poprawa.

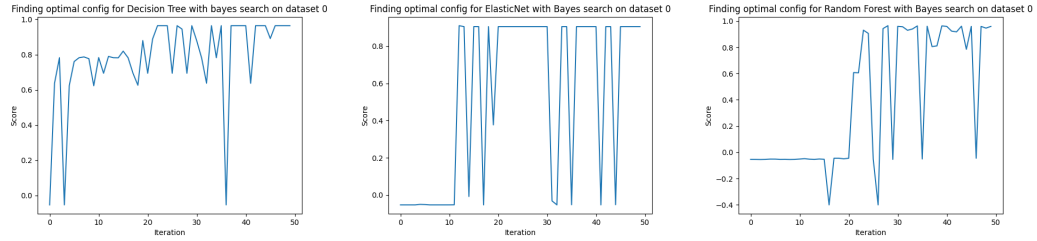


(a) (b) (c)

Grafika 1: Szukanie najlepszych wartości hiperparametrów z użyciem Random Search w zależności od iteracji dla (a) Decision Tree (b) ElasticNet (c) Random Forest

2.2.2 Sampling z optymalizacją Bayes'a

Sampling punktów metodą Bayes'a w celu uzyskania najlepszych hiperparametrów dla danego zbiory danych, daje ciekawsze wyniki, dlatego zostały omówione osobno dla każdego modelu.



(a) (b) (c)

Grafika 2: Przeszukiwanie przestrzeni hiperparametrów z użyciem optymalizacji Bayes'a dla modelu (a) Decision Tree (b) ElasticNet (c) Random Forest

Decision Tree

Na każdym zbiorze danych stabilizacja nie następuje w podobnych przedziałach (momentach w iteracji). Jednak zauważyć można iż po wykonaniu 20-30 kroków iteracji tej metody potrafiliśmy uzyskać optymalną konfigurację.

ElasticNet

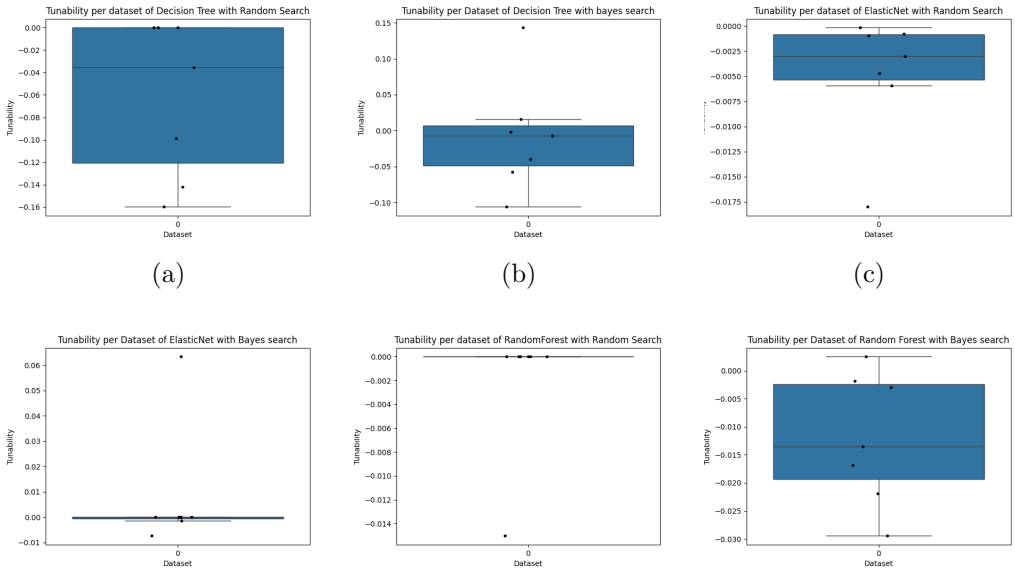
W zależności od zbioru danych, obserwujemy stabilizacje wyników po 10, maksymalnie 15 iteracjach.

Random Forest

W zależności od zbioru danych, obserwujemy stabilizacje wyników po ok. maksymalnie 20 iteracjach optymalizacji.

3 Analiza tunowalności modeli

Tunowalność modeli obliczono zgodnie z wzorem podanym w „Tunability: Importance of Hyperparameters of Machine".



Grafika 3: Tunowalność na każdym z zbiorów danych modelu (a) Decision Tree z Random Search (b) Decision Tree z optymalizacją Bayes'a (c) ElasticNet z Random Search (d) ElasticNet z optymalizacją Bayes'a (e) Random Forest z Random Search (f) Random Forest z optymalizacją Bayes'a

3.1 Decision Tree

Z przeprowadzonego eksperymentu można zauważyć, że model Decision Tree jest podatny na tunowania w zauważalnym stopniu. Co ciekawe, Random Search sprawia wrażenie, że radzi sobie lepiej z tunowaniem, niż optymalizacja Bayes'a.

3.2 Elastic net

Z danych uzyskanych w eksperymencie wynika, że ElasticNet nie jest praktycznie w ogóle podatny na tunowanie, ani Random Search'em, ani optymalizacją

Bayes'a.

3.3 Random Forest

Z danych eksperymentalnych wynika, że Random Forest jest w dość dużym stopniu podatny na tunowanie z wykorzystaniem optymalizacji Bayes'a. Natomiast model nie jest podatny w ogóle na tunowanie z wykorzystaniem Random Search.

3.4 Sampling bias

W celu sprawdzenia sampling bias, przeliczyliśmy wszystkie dane dla różnych wartości „random_state". Z naszych obserwacji wynika, że sampling bias występuje. Z niewielkiej ilości prób z różnymi „random seed"można odnieść wrażenie, że Random Search jest w większym stopniu podatny na tzw. „sampling bias"w porównaniu z optymalizacją Bayesa.

3.5 Wnioski

- Różne modele są w różnym stopniu tunowalne niezależnie od metody przeszukiwania przestrzeni hiperparametrów. Najbardziej podatnym na tunowanie modelem jest Random Forest, a najmniej ElasticNet.
- Pomimo stosowania złożonych operacji wyznaczania kolejnych punktów metoda optymalizacji Bayes'a nie daje zawsze znacznie lepszych wyników od metody Random Search, jest za to zdecydowanie bardziej czasochłonna obliczeniowo.

4 Podsumowanie

Biorąc pod uwagę wyniki powyższych eksperymentów można dojść do wniosku, że proces wyznaczania optymalnych parametrów jest skomplikowany oraz w wysokim stopniu zależy zarówno od metody przeszukiwania przestrzeni jaki i wyborze próbek („sampling bias"ma duże znaczenie na wybór optymalnych hiperparametrów), które to w dużym stopniu wpływają na wynik eksperymentu.

5 Dodatki

5.1 Przestrzenie hiperparametrów

Decision Tree

```
model__ccp_alpha=[0,1]
model__max_depth=[1,30]
model__min_samples_split=[2,60]
model__min_samples_leaf=[1,60]
```

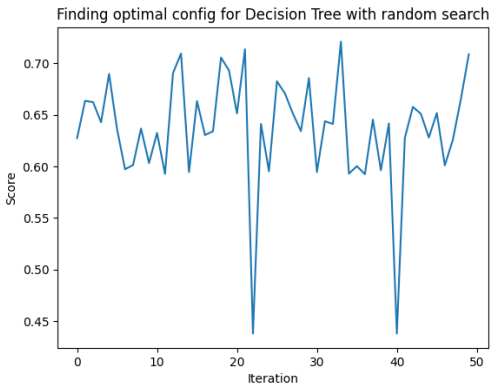
ElasticNet

```
model__alpha=[2**-10,2**10]
model__l1_ratio=[0,1]
model__min_samples_split=[2,60]
model__min_samples_leaf=[1,60]
```

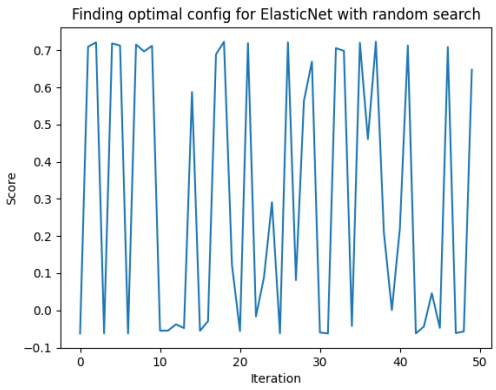
Random Forest

```
model__min_samples_split=[2,528]
model__min_samples_leaf=[2,10]
model__min_samples_leaf=[1,60]
model__n_estimators=[1,2000]
model__max_samples_values=[0.1,1.0]
model__max_features_values=[1,14]
```

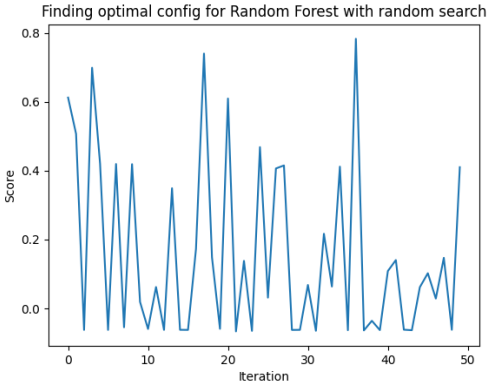
5.2 Szukanie optymalnych hiperparametrów



(a)



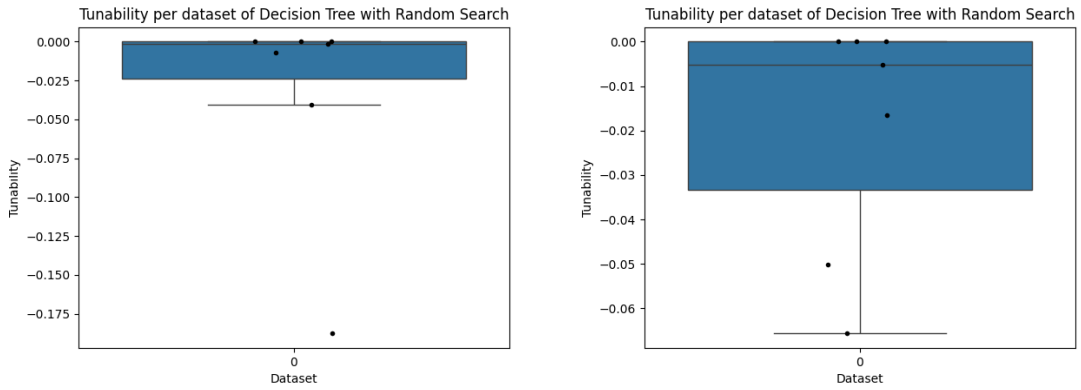
(b)



(c)

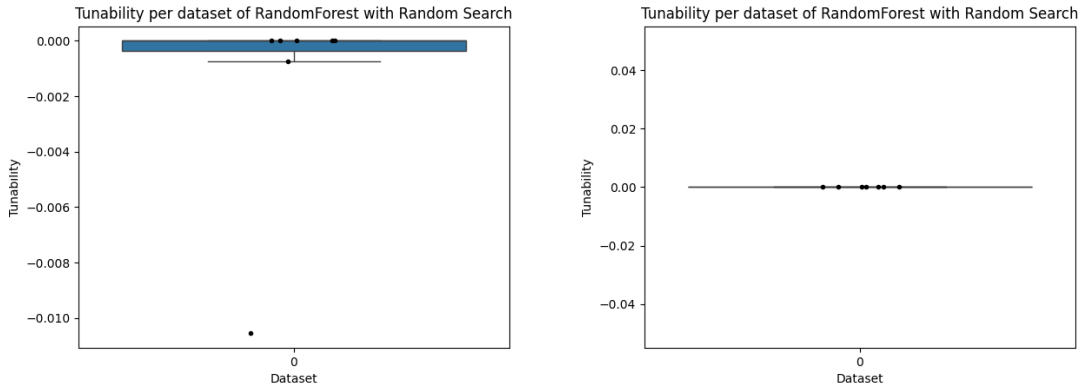
Grafika 4: Szukanie hiperparametrów, dających średnio najlepsze wyniki

5.3 Sampling bias



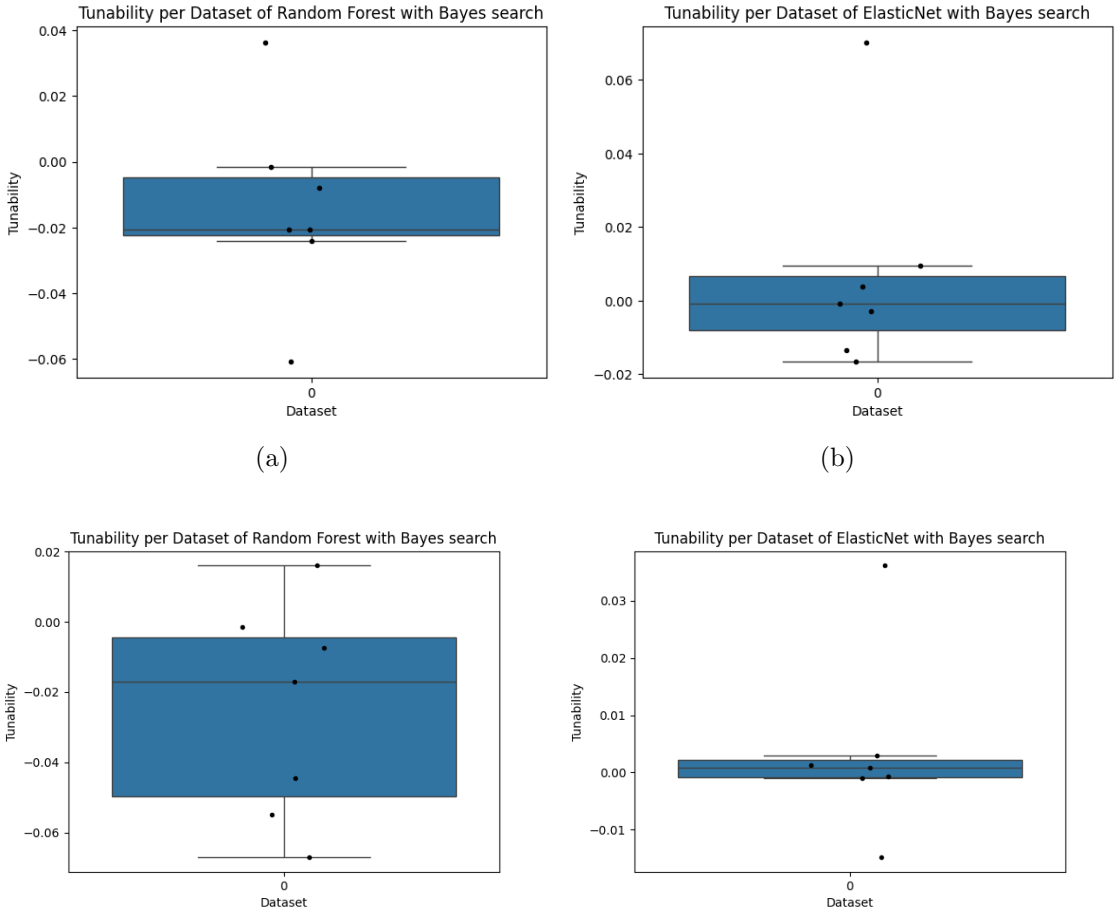
(a) (b)

Grafika 5: Sampling bias dla Random Search na podstawie Decision Tree z wartością seed (a) 321 (b) 5678



(a) (b)

Grafika 6: Sampling bias dla Random Search na podstawie Random Forest z wartością seed (a) 111 (b) 4734



Grafika 7: Sampling bias dla Bayes Search na podstawie Random Forest i ElasticNet z wartością seed (a) 420 (b) 420 (c) 321 (d) 321

5.4 Zbiory danych

Linki do wykorzystanych zbiorów danych:

- `https://www.openml.org/search?type=data&id=43308&sort=runs&status=active`
- `https://www.openml.org/search?type=data&status=active&id=44994`
- `https://www.openml.org/search?type=data&id=44223&sort=runs&status=active`
- `https://www.openml.org/search?type=data&id=43660&sort=runs&status=active`
- `https://www.openml.org/search?type=data&status=active&id=666`
- `https://www.openml.org/search?type=data&status=active&id=531`

- https://www.openml.org/search?type=data&status=active&id=42367