

Tunowalność hiperparametrów

Julia Strzelczyk, Maksymilian Tabian, Łukasz Wyszomierski

18 listopada 2025

1 Wprowadzenie

1.1 Cel projektu

Celem projektu była analiza tunowalności hiperparametrów trzech wybranych algorytmów uczenia maszynowego na co najmniej czterech zbiorach danych. Do tunowania modeli należało wykorzystać min. dwie techniki losowania punktów.

1.2 Zbiory danych i preprocessing

Do projektu zostało wybranych 8 zbiorów danych pochodzących z platformy OpenML, które dotyczyły problemów klasyfikacyjnych: wdbc, ilpd, blood-transfusion-service-center, haberman, heart-statlog, hepatitis, kc2 i thoracic-surgery. Są to zbiory danych o różnorodnej trudności i charakterystyce, co potwierdzają wyniki otrzymanych eksperymentów.

Każdy ze zbiorów został podzielony na część treningową i testową w stosunku 4:1 przy zachowaniu struktury klas. Zmienna docelowa została zakodowana przy użyciu LabelEncoder. Dla wszystkich modeli zastosowano jednakowy schemat wstępnego przetwarzania danych. W przypadku zmiennych numerycznych brakujące wartości uzupełniano średnią, a następnie dokonano ich standaryzacji. Brakujące wartości w zmiennych kategorycznych zostały zaimputowane za pomocą najczęściej występującej wartości. Kategorie były kodowane przy użyciu OneHotEncoder.

2 Modele i tunowanie hiperparametrów

2.1 Wybrane modele

Do analizy wybrano trzy algorytmy opierające się na idei komitetów drzew, będące często podstawowymi wyborami przy problemie klasyfikacji binarnej dla danych tabelarycznych - Random Forest, XGBoost oraz LightGBM. Prze prowadzone z ich pomocą eksperymenty mają charakter wielowymiarowy. Po pierwsze, pozwoliły zweryfikować jak modele te radzą sobie na zróżnicowanym zestawie danych tabelarycznych. Jednocześnie dla każdego z tych modeli zbadano tunowalność, co pozwoliło ocenić jak efektywny jest dla nich proces doboru hiperparametrów.

2.2 Rozważane hiperparametry

W eksperymencie skupiono się na hiperparametrach mających największy wpływ na wydajność predykcyjną modeli, zgodnie z motywacjami wynikającymi z literatury:

- **Random Forest**

Na podstawie artykułu Tunability [1] wybrano trzy kluczowe parametry: `max_features` (odpowiednik `mtry`) regulujący korelację między drzewami, `max_samples` (odpowiednik `sample_fraction`) kontrolujący stopień losowości bootstrapu oraz `n_estimators` (odpowiednik `num_trees`), który badano w zakresie 100–300, aby zapewnić stabilność predykcji przy zachowaniu rozsądnego czasu treningu.

- **XGBoost**

Na podstawie artykułu Tunability [1], kluczowym hiperparametrem jest `learning_rate` (odpowiednik `eta`). Jest on ściśle powiązany z liczbą estymatorów `n_estimators`, który również sprawdzano. Dodatkowo sprawdzano parametr `booster`, porównując standardowe drzewa (gbtree) z metodą wykorzystującą dropout (dart).

- **LightGBM**

Zgodnie z dokumentacją biblioteki¹, głównym parametrem kontrolującym złożoność modelu jest `num_leaves` (LGBM buduje drzewa leaf-wise, a nie level-wise), dlatego to na nim skupiono uwagę zamiast na głębokości drzewa. Podobnie jak w XGBoost, kluczową rolę odgrywa para `learning_rate` oraz `n_estimators`.

¹<https://lightgbm.readthedocs.io/en/stable/Parameters-Tuning.html>

Rozważane hiperparametry wraz z zakresami przedstawiono w tabeli:

Model	Hiperparametr	Badany Zakres / Wartości	Wartość Domyślna
Random Forest	<code>max_features</code>	0.1 – 1.0 (float)	"sqrt" (\approx pierwiastek z liczby cech)
	<code>max_samples</code>	0.5 – 1.0	<code>None</code> (wszystkie próbki)
	<code>n_estimators</code>	100 – 300	100
XGBoost	<code>learning_rate</code>	0.01 – 0.4	0.3
	<code>n_estimators</code>	100 – 300	100
	<code>booster</code>	[<code>'gbtree'</code> , <code>'dart'</code>]	<code>'gbtree'</code>
LightGBM	<code>num_leaves</code>	10 – 60	31
	<code>learning_rate</code>	0.01 – 0.4	0.1
	<code>n_estimators</code>	100 – 300	100

2.3 Wybrane metody samplingu

W projekcie zastosowano trzy metody losowania punktów: Grid Search i Random Search z pakietu scikit-learn oraz Bayes Search z pakietu skopt, wykorzystujący optymalizację bayesowską. Modele były oceniane za pomocą miary AUC. Jakość każdej kombinacji hiperparametrów była sprawdzana za pomocą 5-krotnej kroswalidacji na zbiorze treningowym. W przypadku metod Random Search i Bayes Search wykonano po 125 iteracji. Dla metody Grid Search wartości hiperparametrów zostały dobrane równomiernie w badanych zakresach, tak by liczba sprawdzanych kombinacji była zbliżona do liczby iteracji sprawdzanych w metodach Random i Bayes Search.

3 Wyniki tunowania

3.1 Nowe wartości domyślne hiperparametrów

Nowe domyślne wartości hiperparametrów wyznaczono zgodnie z metodologią przedstawioną w artykule [1] tzn.

$$\theta^* = \arg \min_{\theta \in \Theta} g(R^{(1)}(\theta), \dots, R^{(m)}(\theta)),$$

gdzie jako funkcję g przyjęto średnią, natomiast jako funkcję ryzyka $R(\theta)$ przyjęto ujemną wartość metryki AUC.

W praktyce oznacza to, że poszukiwano jednej, uniwersalnej konfiguracji hiperparametrów, która maksymalizowała wynik AUC uzyskiwany w kroswalidacji, uśredniony po wszystkich analizowanych zbiorach danych (tj. wybrano ten zestaw parametrów, który dał najwyższe średnie `mean_test_score` globalnie dla wszystkich datasetów). Otrzymane wartości przedstawiono w tabeli:

Random Forest			
Metoda tunowania	<code>max_features</code>	<code>max_samples</code>	<code>n_estimators</code>
Grid Search	1.0000	0.8000	150
Random Search	0.3527	0.5122	148
XGBoost			
Metoda tunowania	<code>booster</code>	<code>learning_rate</code>	<code>n_estimators</code>
Grid Search	dart	0.0500	100
Random Search	gbtree	0.0281	150
LightGBM			
Metoda tunowania	<code>learning_rate</code>	<code>n_estimators</code>	<code>num_leaves</code>
Grid Search	0.0100	300	12
Random Search	0.0155	180	17

3.2 Tunowalność algorytmów

Zgodnie z metodologią opisaną w artykule [1], tunowalność każdego z rozważanych modeli na j -tym zbiorze danych zdefiniowano jako różnicę między ryzykiem dla wartości domyślnych a ryzykiem dla wartości optymalnych:

$$d^{(j)} = R^{(j)}(\theta_{default}) - R^{(j)}(\theta_{opt}^{(j)}). \quad (1)$$

Analogicznie do poprzednich sekcji, jako funkcję ryzyka R przyjęto ujemną wartość metryki AUC ($-AUC$).

Szczegółowe rezultaty dla każdego modelu i metody optymalizacji z podziałem na zbiór danych zestawiono w tabelach: [1]–[9]. Z kolei w tabeli [10] przedstawiono uśrednione wyniki, pozwalające ocenić tunowalność danego algorytmu. Aby zapewnić pełny obraz skuteczności optymalizacji, wyniki w tej tabeli podzielono na dwie kategorie:

- **Wyniki z kroswalidacją (Tun.O-CV)**- tunowalność obliczono jako różnicę między wynikiem uzyskanym dla nowo wyznaczonych domyślnych hiperparametrów, a najlepszym wynikiem znalezionym przez algorytm w procesie kroswalidacji na danym zbiorze.
- **Wyniki na zbiorze testowym**- tunowalność obliczono porównując na niezależnym zbiorze testowym najlepsze hiperparametry znalezione w kroswalidacji z domyślnymi wartościami:
 1. **Tun.P** - tunowalność względem wbudowanych wartości domyślnych danej biblioteki,
 2. **Tun.O** - tunowalność względem nowo wyznaczonych wartości domyślnych.

Należy zaznaczyć, że w przypadku analizy metody Bayes Search, jako punkt odniesienia dla nowych wartości domyślnych przyjęto wartości wyznaczone wcześniej metodą Grid Search.

Otrzymane wyniki można rozpatrywać w kilku płaszczyznach. Po pierwsze zachodzi zmniejszenie tunowalności po ustaleniu nowych domyślnych hiperparametrów dla metody losowania punktów za pomocą Random Search (Tun.P > Tun.O). Własność ta zachodzi dla wszystkich modeli. Dodatkowo dla modeli XGBoost i LightGBM podobne wnioski zachodzą przy losowaniu punktów metodą Grid Search. Modele boostingowe (XGBoost, LightGBM) dla domyślnych hiperparametrów wykazują wyższy potencjał poprawy wyników poprzez tunowanie niż Random Forest (Tun.P), jednak warto pamiętać że średni wynik obarczony jest sporym błędem standardowym.

3.3 Wpływ wielkości zbioru danych na wyniki tunowania

W celu zbadania, jak liczba dostępnych obserwacji wpływa na tunowalność modeli, przeprowadzono dodatkowy eksperyment, w którym procedurę wyznaczania optymalnych hiperparametrów powtórzono dla podzbiorów o licznosciach 50%, 75% oraz 100% oryginalnego zbioru treningowego. Dla każdego modelu wyznaczono tunowalność przy zastosowaniu metod Random Search oraz Grid Search. Uzyskane wyniki przedstawiono na wykresach [1], [2] oraz [3]. Widać na nich, że na ogół tunowalność wyznaczona na podstawie zbiorów zawierających 75% i 100% obserwacji nie różni się znacząco, natomiast w przypadku wartości wyznaczonych na podstawie podzbioru liczącego 50% obserwacji widoczne są większe różnice.

3.4 Stabilność wyników optymalizacji

W celu oceny szybkości zbieżności metod tunowania hiperparametrów sprawdzono, po ilu iteracjach metody Bayes Search i Random Search wyniki AUC ulegają stabilizacji. Otrzymane wyniki przedstawiają wykresy [7] oraz [8]. Na ich podstawie widać, że w przypadku obu metod po około 20 iteracjach otrzymujemy stabilne rezultaty.

3.5 Rozkład różnic pomiędzy miarą AUC otrzymaną dla wyznaczonej domyślnej konfiguracji a miarami AUC uzyskanymi dla innych testowanych konfiguracji

Przeprowadzono analizę rozkładów różnic pomiędzy wartościami miary AUC otrzymanymi dla testowanych konfiguracji a wartościami AUC uzyskanymi dla wyznaczonych domyślnych konfiguracji ($AUC_{tested_config} - AUC_{new_default}$). W przypadku metody Bayes Search jako konfigurację domyślną przyjęto tę wyznaczoną dla Grid Search. Wyniki zostały przedstawione na wykresach [4], [5] i [6].

Wyniki różnią się w zależności od zbioru danych oraz modelu, jednak w większości przypadków wartości AUC uzyskane dla testowanych konfiguracji są niższe niż dla nowych domyślnych hiperparametrów. Co więcej, można zauważać, że rozkłady dla trzech metod samplingu są do siebie dość podobne, lecz wyniki uzyskane za pomocą metody Grid Search charakteryzują się największym rozstępem międzykwartylowym, w szczególności dla modelu LightGBM.

3.6 Bias sampling

Na podstawie porównania rozkładów tunowalności, które zostały przedstawione na wykresach [9] i [10], można stwierdzić, że technika losowania punktów nie wpływa w sposób istotny na wyniki, gdyż rozkłady dla trzech metod samplingu są do siebie zbliżone.

3.7 Porównanie różnic wyników dla różnych technik losowania hiperparametrów przy pomocy testów statystycznych

W celu porównania różnic dla poszczególnych technik losowania hiperparametrów posłużyono się nieparametryczną analizą schematów blokowych. Dla każdego z rozważanych modeli stworzono następujący schemat blokowy:

	Metoda losowania hiperparametrów			
	Grid Search	Random Search	Bayes Search	Default
zbiór 1	AUC_{11}	AUC_{12}	AUC_{13}	AUC_{14}
zbiór 2	AUC_{21}	AUC_{22}	AUC_{23}	AUC_{24}
\vdots	\vdots	\vdots	\vdots	\vdots
zbiór 8	AUC_{81}	AUC_{82}	AUC_{83}	AUC_{84}

AUC_{ij} oznacza tutaj AUC na zbiorze numer i dla modelu o hiperparametrach wybranych j -tą metodą. W schemacie uwzględniono również wartości AUC na poszczególnych zbiorach osiągnięte dla nowych domyślnych wartości hiperparametrów wyznaczonych dla metody Grid Search zgodnie z uprzednio zaprezentowaną metodologią.

Po stworzeniu odpowiedniego schematu blokowego dla każdego z rozważanych modeli, przeprowadzono test Friedmana do weryfikacji następującej hipotezy:

$$\begin{cases} H_0 : \text{rozkłady AUC dla wszystkich metod tunowania hiperparametrów są jednakowe} \\ H_1 : \neg H_0. \end{cases}$$

Otrzymane p-wartości prezentuje tabela:

Model	p-wartość
Random Forest	0.2898
XGBoost	0.3476
LightGBM	0.4153

Wszystkie p-wartości są większe niż 0.05 zatem dla żadnego z rozważanych modeli nie ma podstaw do odrzucenia hipotezy zerowej o równości rozkładów AUC dla poszczególnych metod tunowania hiperparametrów.

4 Tabele i wykresy

Random Forest + Grid Search					
zbiór danych	AUC dla wbudowanych wartości domyślnych	AUC dla nowych wartości domyślnych	AUC dla optymalnych hiperparametrów	$d_P^{(j)}$	$d_O^{(j)}$
blood-transfusion-service-center	0.685551	0.703582	0.696516	0.010965	-0.007066
haberman	0.557065	0.555027	0.572011	0.014946	0.016984
heart-statlog	0.915972	0.897917	0.937500	0.021528	0.039583
hepatitis	0.856667	0.833333	0.806667	-0.050000	-0.026667
ilpd	0.772325	0.772147	0.815025	0.042700	0.042877
kc2	0.771906	0.756846	0.776287	0.004381	0.019441
thoracic-surgery	0.613839	0.650893	0.619643	0.005804	-0.031250
wdbc	0.990575	0.990245	0.988757	-0.001819	-0.001488

Tabela 1: Porównanie tunowalności wyznaczonej na zbiorach testowych na podstawie wbudowanych domyślnych wartości hiperparametrów ($d_P^{(j)}$) oraz nowych domyślnych wartości ($d_O^{(j)}$).

Random Forest + Random Search					
zbiór danych	AUC dla wbudowanych wartości domyślnych	AUC dla nowych wartości domyślnych	AUC dla optymalnych hiperparametrów	$d_P^{(j)}$	$d_O^{(j)}$
blood-transfusion-service-center	0.685551	0.689693	0.689693	0.004142	0.000000
haberman	0.557065	0.561821	0.561141	0.004076	-0.000679
heart-statlog	0.915972	0.915972	0.918750	0.002778	0.002778
hepatitis	0.856667	0.843333	0.856667	0.000000	0.013333
ilpd	0.772325	0.798724	0.796775	0.024451	-0.001949
kc2	0.771906	0.768894	0.771632	-0.000274	0.002738
thoracic-surgery	0.613839	0.636161	0.649107	0.035268	0.012946
wdbc	0.990575	0.991071	0.989253	-0.001323	-0.001819

Tabela 2: Porównanie tunowalności wyznaczonej na zbiorach testowych na podstawie wbudowanych domyślnych wartości hiperparametrów ($d_P^{(j)}$) oraz nowych domyślnych wartości ($d_O^{(j)}$).

Random Forest + Bayes Search					
zbiór danych	AUC dla wbudowanych wartości domyślnych	AUC dla nowych wartości domyślnych	AUC dla optymalnych hiperparametrów	$d_P^{(j)}$	$d_O^{(j)}$
blood-transfusion-service-center	0.685551	0.703582	0.695663	0.010112	-0.007919
haberman	0.557065	0.555027	0.561141	0.004076	0.006114
heart-statlog	0.915972	0.897917	0.932639	0.016667	0.034722
hepatitis	0.856667	0.833333	0.843333	-0.013333	0.010000
ilpd	0.772325	0.772147	0.788802	0.016478	0.016655
kc2	0.771906	0.756846	0.763417	-0.008488	0.006572
thoracic-surgery	0.613839	0.650893	0.691071	0.077232	0.040179
wdbc	0.990575	0.990245	0.990575	0.000000	0.000331

Tabela 3: Porównanie tunowalności wyznaczonej na zbiorach testowych na podstawie wbudowanych domyślnych wartości hiperparametrów ($d_P^{(j)}$) oraz nowych domyślnych wartości (wyznaczonych dla Grid Search) ($d_O^{(j)}$).

XGBoost + Grid Search					
zbiór danych	AUC dla wbudowanych wartości domyślnych	AUC dla nowych wartości domyślnych	AUC dla optymalnych hiperparametrów	$d_P^{(j)}$	$d_O^{(j)}$
blood-transfusion-service-center	0.709064	0.726243	0.726243	0.017178	0.000000
haberman	0.514946	0.582880	0.582880	0.067935	0.000000
heart-statlog	0.909722	0.904167	0.915278	0.005556	0.011111
hepatitis	0.780000	0.846667	0.860000	0.080000	0.013333
ilpd	0.761871	0.779589	0.779589	0.017718	0.000000
kc2	0.698521	0.748631	0.732475	0.033954	-0.016156
thoracic-surgery	0.649107	0.612054	0.636607	-0.012500	0.024554
wdbc	0.990079	0.989749	0.989418	-0.000661	-0.000331

Tabela 4: Porównanie tunowalności wyznaczonej na zbiorach testowych na podstawie wbudowanych domyślnych wartości hiperparametrów ($d_P^{(j)}$) oraz nowych domyślnych wartości ($d_O^{(j)}$).

XGBoost + Random Search					
zbiór danych	AUC dla wbudowanych wartości domyślnych	AUC dla nowych wartości domyślnych	AUC dla optymalnych hiperparametrów	$d_P^{(j)}$	$d_O^{(j)}$
blood-transfusion-service-center	0.709064	0.724659	0.719298	0.010234	-0.005361
haberman	0.514946	0.574728	0.574728	0.059783	0.000000
heart-statlog	0.909722	0.905556	0.913889	0.004167	0.008333
hepatitis	0.780000	0.853333	0.846667	0.066667	-0.006667
ilpd	0.761871	0.773210	0.753012	-0.008859	-0.020198
kc2	0.698521	0.741238	0.740964	0.042442	-0.000274
thoracic-surgery	0.649107	0.604911	0.643750	-0.005357	0.038839
wdbc	0.990079	0.989087	0.987765	-0.002315	-0.001323

Tabela 5: Porównanie tunowalności wyznaczonej na zbiorach testowych na podstawie wbudowanych domyślnych wartości hiperparametrów ($d_P^{(j)}$) oraz nowych domyślnych wartości ($d_O^{(j)}$).

XGBoost + Bayes Search					
zbiór danych	AUC dla wbudowanych wartości domyślnych	AUC dla nowych wartości domyślnych	AUC dla optymalnych hiperparametrów	$d_P^{(j)}$	$d_O^{(j)}$
blood-transfusion-service-center	0.709064	0.726243	0.725877	0.016813	-0.000365
haberman	0.514946	0.582880	0.587636	0.072690	0.004755
heart-statlog	0.909722	0.904167	0.909722	0.000000	0.005556
hepatitis	0.780000	0.846667	0.846667	0.066667	0.000000
ilpd	0.761871	0.779589	0.783133	0.021262	0.003546
kc2	0.698521	0.748631	0.756298	0.057777	0.007667
thoracic-surgery	0.649107	0.612054	0.666071	0.016964	0.054018
wdbc	0.990079	0.989749	0.989749	-0.000331	0.000000

Tabela 6: Porównanie tunowalności wyznaczonej na zbiorach testowych na podstawie wbudowanych domyślnych wartości hiperparametrów ($d_P^{(j)}$) oraz nowych domyślnych wartości (wyznaczonych dla Grid Search) ($d_O^{(j)}$).

LightGBM + Grid Search					
zbiór danych	AUC dla wbudowanych wartości domyślnych	AUC dla nowych wartości domyślnych	AUC dla optymalnych hiperparametrów	$d_P^{(j)}$	$d_O^{(j)}$
blood-transfusion-service-center	0.733065	0.740132	0.737329	0.004264	-0.002802
haberman	0.565217	0.584239	0.584239	0.019022	0.000000
heart-statlog	0.923611	0.919444	0.930556	0.006944	0.011111
hepatitis	0.740000	0.846667	0.880000	0.140000	0.033333
ilpd	0.783841	0.759036	0.776045	-0.007796	0.017009
kc2	0.734666	0.749726	0.744524	0.009858	-0.005203
thoracic-surgery	0.606696	0.588839	0.628571	0.021875	0.039732
wdbc	0.990079	0.988757	0.992063	0.001984	0.003307

Tabela 7: Porównanie tunowalności wyznaczonej na zbiorach testowych na podstawie wbudowanych domyślnych wartości hiperparametrów ($d_P^{(j)}$) oraz nowych domyślnych wartości ($d_O^{(j)}$).

LightGBM + Random Search					
zbiór danych	AUC dla wbudowanych wartości domyślnych	AUC dla nowych wartości domyślnych	AUC dla optymalnych hiperparametrów	$d_P^{(j)}$	$d_O^{(j)}$
blood-transfusion-service-center	0.733065	0.739035	0.739035	0.005970	0.000000
haberman	0.565217	0.589674	0.575408	0.010190	-0.014266
heart-statlog	0.923611	0.916667	0.918056	-0.005556	0.001389
hepatitis	0.740000	0.853333	0.786667	0.046667	-0.066667
ilpd	0.783841	0.772502	0.777193	-0.012048	0.004691
kc2	0.734666	0.746166	0.746166	0.011501	0.000000
thoracic-surgery	0.606696	0.590625	0.623214	0.016518	0.032589
wdbc	0.990079	0.987434	0.989087	-0.000992	0.001653

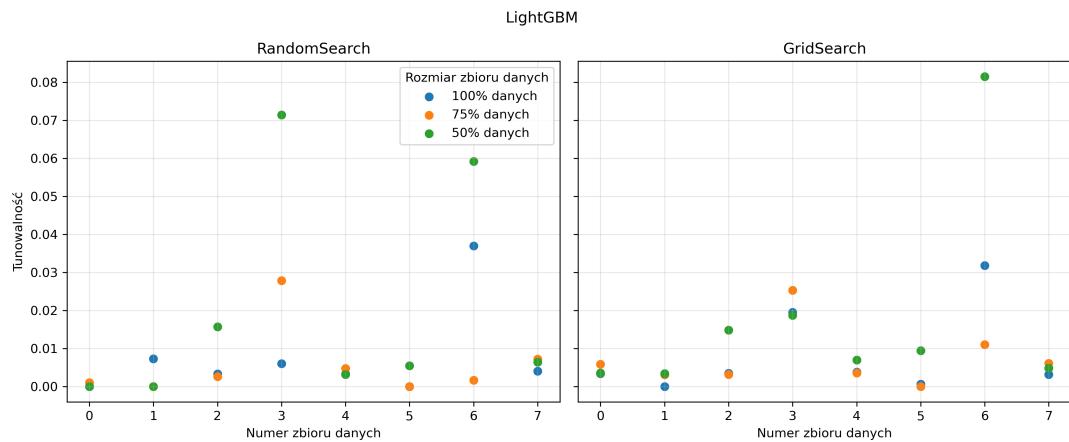
Tabela 8: Porównanie tunowalności wyznaczonej na zbiorach testowych na podstawie wbudowanych domyślnych wartości hiperparametrów ($d_P^{(j)}$) oraz nowych domyślnych wartości ($d_O^{(j)}$).

LightGBM + Bayes Search					
zbiór danych	AUC dla wbudowanych wartości domyślnych	AUC dla nowych wartości domyślnych	AUC dla optymalnych hiperparametrów	$d_P^{(j)}$	$d_O^{(j)}$
blood-transfusion-service-center	0.733065	0.740132	0.733187	0.000122	-0.006944
haberman	0.565217	0.584239	0.590353	0.025136	0.006114
heart-statlog	0.923611	0.919444	0.929167	0.005556	0.009722
hepatitis	0.740000	0.846667	0.873333	0.133333	0.026667
ilpd	0.783841	0.759036	0.770376	-0.013466	0.011339
kc2	0.734666	0.749726	0.745619	0.010953	-0.004107
thoracic-surgery	0.606696	0.588839	0.637500	0.030804	0.048661
wdbc	0.990079	0.988757	0.991071	0.000992	0.002315

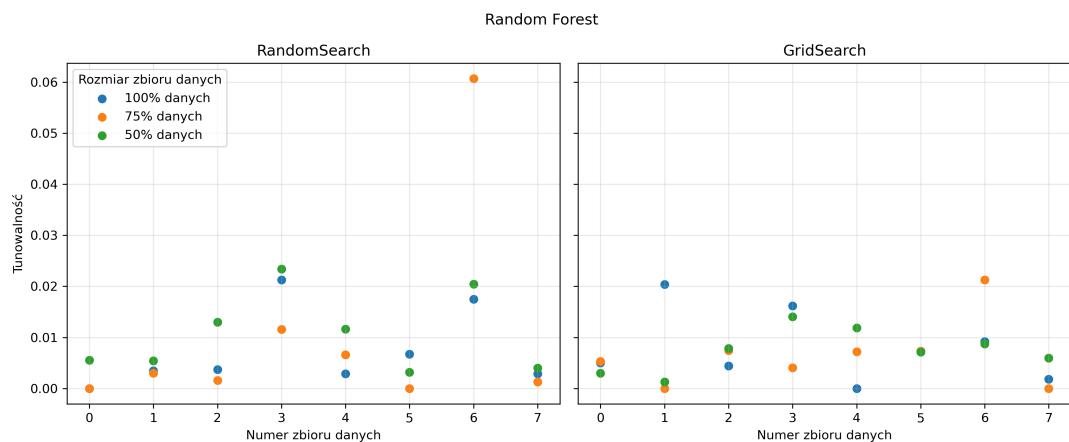
Tabela 9: Porównanie tunowalności wyznaczonej na zbiorach testowych na podstawie wbudowanych domyślnych wartości hiperparametrów ($d_P^{(j)}$) oraz nowych domyślnych wartości (wyznaczonych dla Grid Search) ($d_O^{(j)}$).

Model	Metoda	Tun.O - CV	Tun. P	Tun.O
Random Forest	GridSearch	0.0080 ± 0.0025	0.0061 ± 0.0093	0.0066 ± 0.0099
	RandomSearch	0.0073 ± 0.0027	0.0086 ± 0.0048	0.0034 ± 0.0022
	BayesSearch	0.0137 ± 0.0035	0.0128 ± 0.0100	0.0133 ± 0.0059
XGBoost	GridSearch	0.0070 ± 0.0035	0.0218 ± 0.0104	0.0061 ± 0.0031
	RandomSearch	0.0068 ± 0.0025	0.0214 ± 0.0092	0.0068 ± 0.0063
	BayesSearch	0.0110 ± 0.0051	0.0315 ± 0.0105	0.0094 ± 0.0065
LightGBM	GridSearch	0.0082 ± 0.0040	0.0245 ± 0.0168	0.0121 ± 0.0060
	RandomSearch	0.0076 ± 0.0043	0.0090 ± 0.0063	-0.0058 ± 0.0099
	BayesSearch	0.0111 ± 0.0037	0.0242 ± 0.0164	0.0117 ± 0.0064

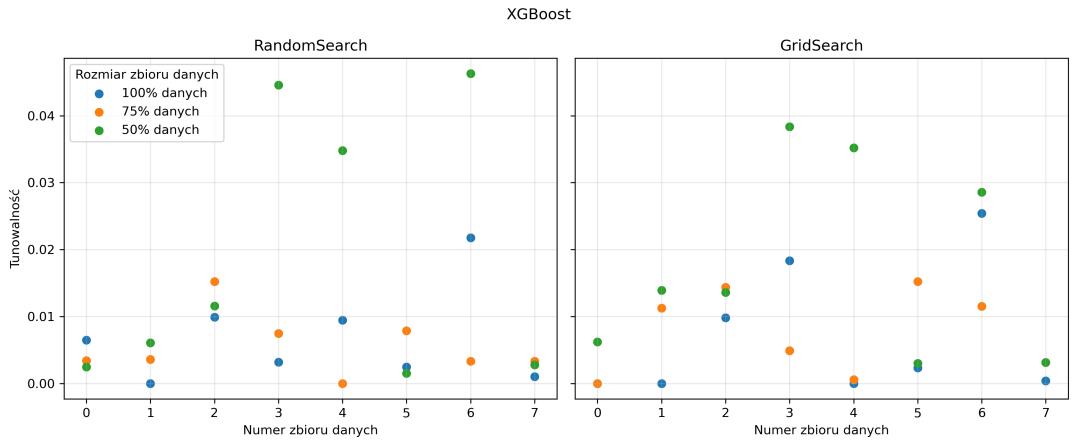
Tabela 10: Tunowalność modeli dla różnych metod losowania punktów (średni AUC ± błąd standaryzowany)



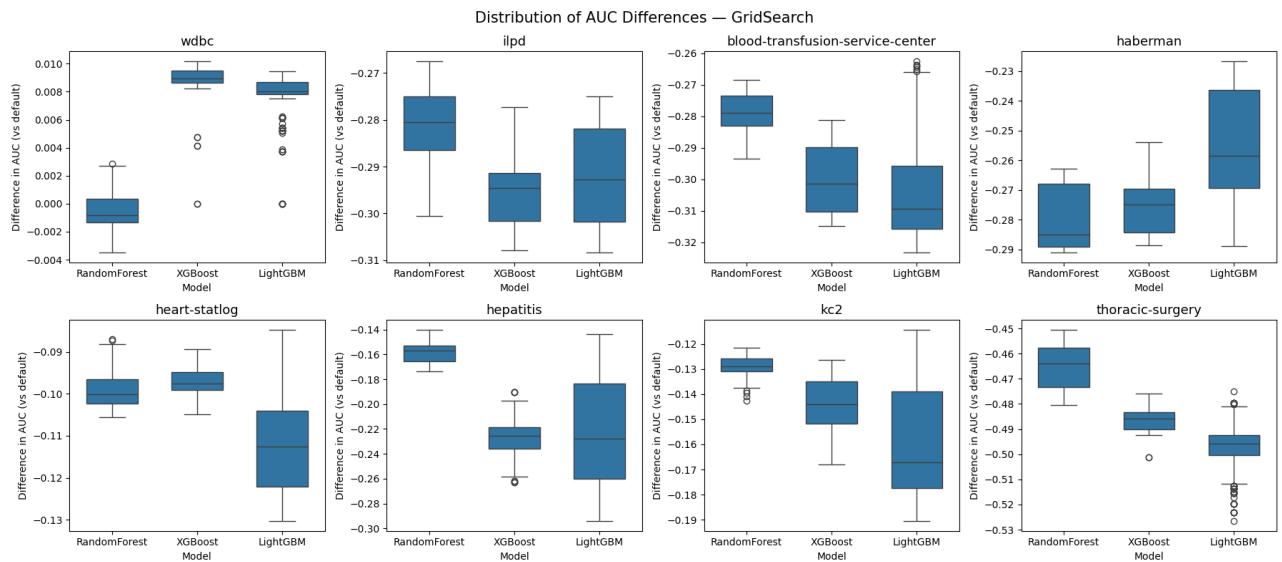
Rysunek 1: Wykres tunowalności dla podzbiorów o liczności 100%, 75%, 50% oryginalnych zbiorów dla modelu LightGBM uzyskanych przy ustalaniu hiperparametrów za pomocą metod RandomSearch i GridSearch.



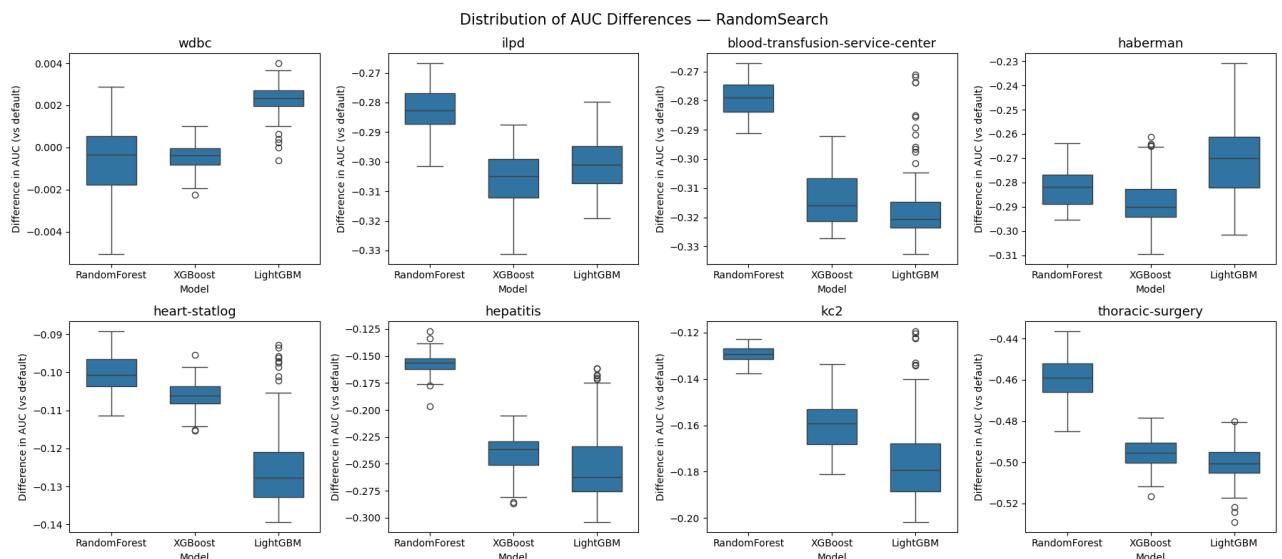
Rysunek 2: Wykres tunowalności dla podzbiorów o liczności 100%, 75%, 50% oryginalnych zbiorów dla modelu Random Forest uzyskanych przy ustalaniu hiperparametrów za pomocą metod RandomSearch i GridSearch.



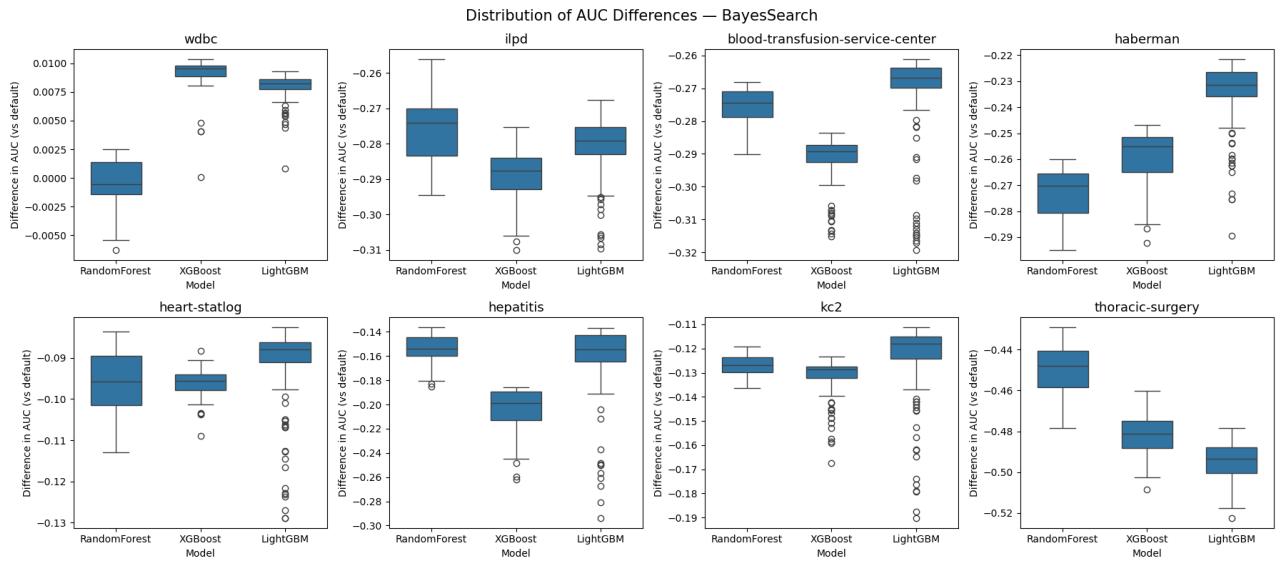
Rysunek 3: Wykres tunowalności dla podzbiorów o liczności 100%, 75%, 50% oryginalnych zbiorów dla modelu XGBoost uzyskanych przy ustalaniu hiperparametrów za pomocą metod RandomSearch i GridSearch.



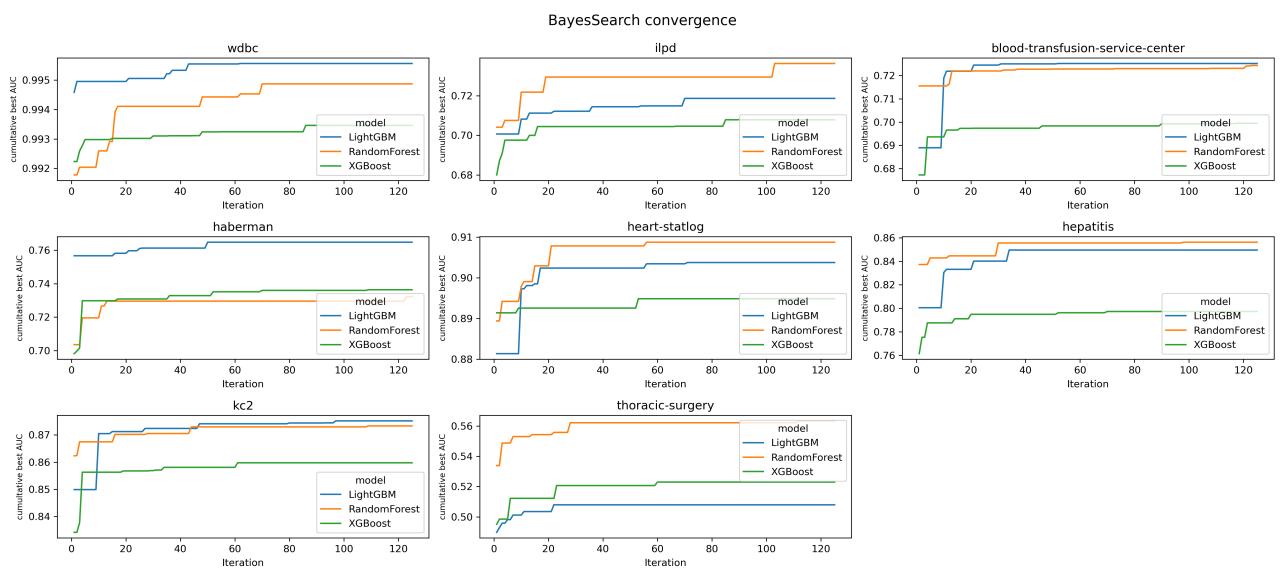
Rysunek 4: Rozkład różnic miary AUC pomiędzy konfiguracją domyślną a testowanymi konfiguracjami uzyskanymi za pomocą GridSearch.



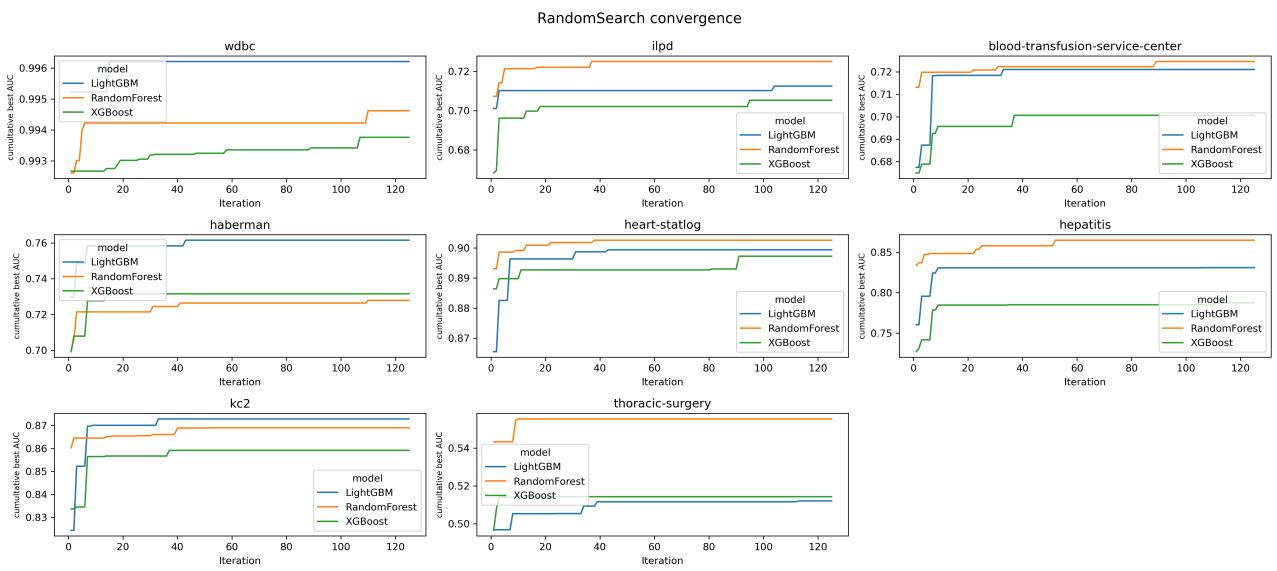
Rysunek 5: Rozkład różnic miary AUC pomiędzy konfiguracją domyślną a testowanymi konfiguracjami uzyskanymi za pomocą RandomSearch.



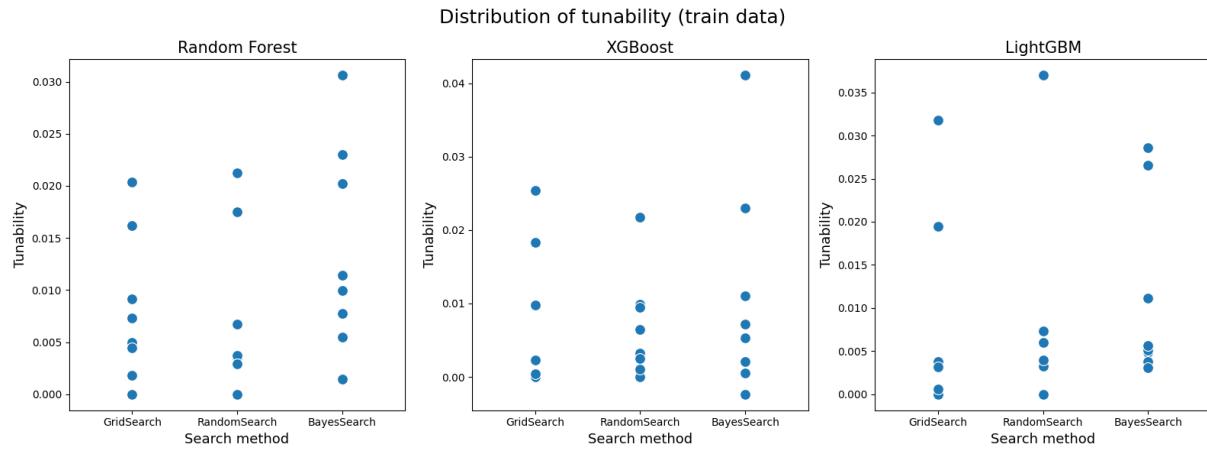
Rysunek 6: Rozkład różnic miary AUC pomiędzy konfiguracją domyślną a testowanymi konfiguracjami uzyskanymi za pomocą BayesSearch.



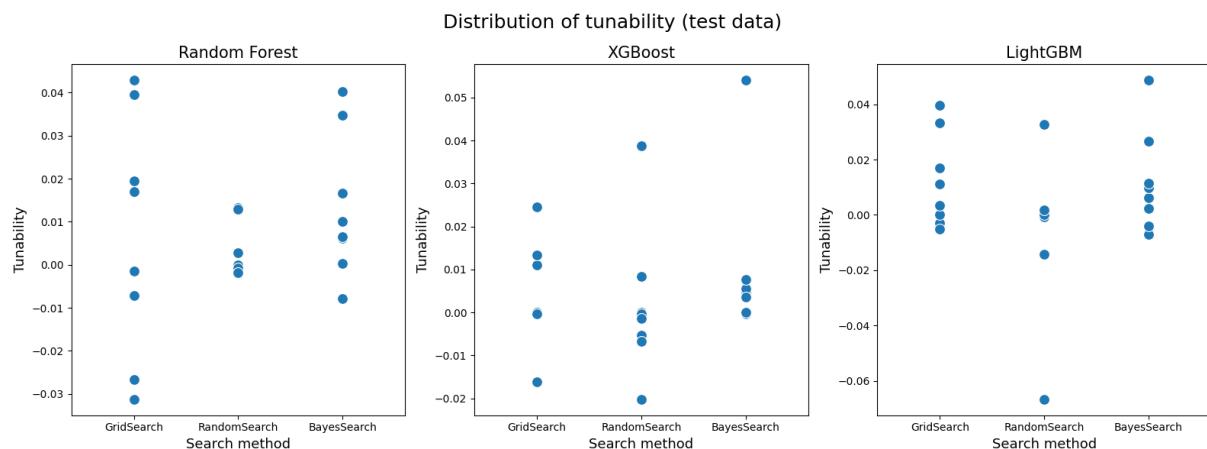
Rysunek 7: Stabilność wyników dla modeli tunowanych metodą Bayes Search.



Rysunek 8: Stabilność wyników dla modeli tunowanych metodą Random Search.



Rysunek 9: Rozkład tunowalności modeli dla danych treningowych.



Rysunek 10: Rozkład tunowalności modeli dla danych testowych.

Literatura

- [1] Philipp Probst, Anne-Laure Boulesteix, Bernd Bischl. Tunability: Importance of Hyperparameters of Machine Learning Algorithms. *Journal of Machine Learning Research*, 2019.