# Analiza tunowalności hiperparametrów wybranych algorytmów uczenia maszynowego

AutoML 2024Z

Zofia Kamińska Mateusz Deptuch Karolina Dunal

# Plan prezentacji

01 Dane

02 Algorytmy

03 Random Search

04 Bayes Optimization

05 Porównanie wyników

06 Podsumowanie

# 01| Dane

**4 zbiory danych do zadania klasyfikacji binarnej**

Platforma *Kaggle*:

- ***Car Purchase Dataset*** *– 1000 wierszy, 5 kolumn,*
- ***Hiring Dataset*** *– 1500 wierszy, 11 kolumn,*
- ***Diabetes Dataset*** *– 768 wierszy, 10 kolumn,*

Platforma *OpenML*:

- ***Banknote Authentication Dataset*** *– 1372 wierszy, 6 kolumn.*

Preprocessing:

- Brak brakujących wartości,
- **Imputacja** wartości dla zbioru *diabetes,*
- **Kodowanie** (zmienne kategoryczne) – `LabelEncoder`,
- **Standaryzacja** (zmienne numeryczne) – `StandardScaler`.

# 02\ Algorytmy

## Regresja Logistyczna

*LogisticRegression* z pakietu *scikit-learn*

## Las Losowy

*RandomForest* z pakietu *scikit-learn*

## XGBoost

*XGBoost* z pakietu *XGBoost*

# 03 | Random Search

Kryterium optymalizacji: **AUC** (Score)

*Siatka* hiperparametrów: **stała** dla każdego zbioru danych,
zróżnicowana w zależności od modelu

Liczba iteracji: **300**

Nowy **default**: najlepsze AUC
(średnia z wyników na wszystkich 4 zbiorach)

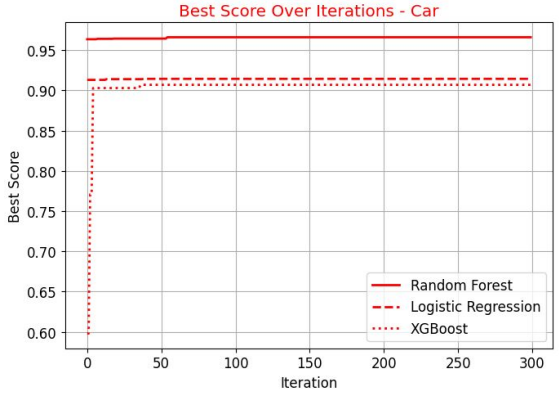
# 03 | Random Search

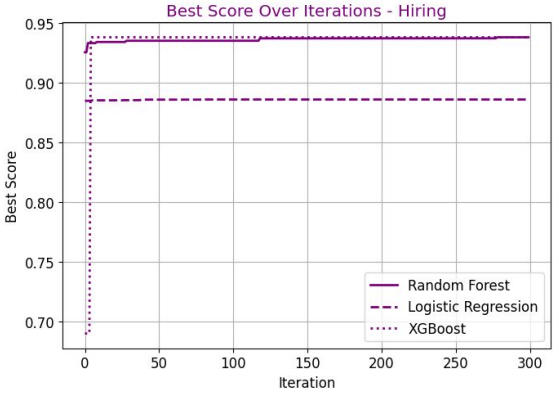
nowy default

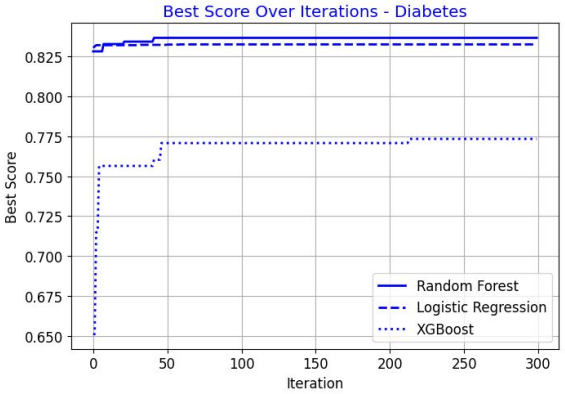
| Regresja logistyczna | XGBoost | Random Forest |
|---|---|---|
| *solver*: saga | *booster*: gbtree | *n_estimators*: 256 |
| *penalty*: elasticnet | *eta*: 0.07906 | *min_samples_split:* 8 |
| *C*: 0.271463 | *max_depth*: 10 | *min_samples_leaf*: 4 |
| *max_iter*: 235 | *min_child_weight*: 7 | *max_features*: 0.2 |
| *tol*: 0.00001 | *subsample*: 0.897959 | *max_depth*: 8 |
| *class_weight*: None | *colsample_bytree*: 0.897959 | *criterion*: log_loss |
| *l1_ratio*: 0.631139 | *gamma*: 0.000029 | *bootstrap*: False |
| | *lambda*: 0.323746 | |
| | *alpha*: 0.000069 | |

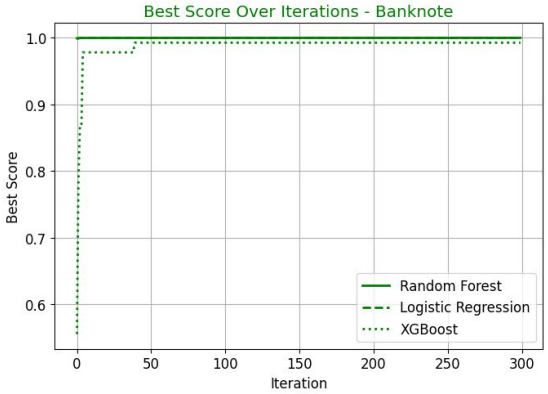
| ***Najlepszy średni* wynik AUC*** | | |
|---|---|---|
| 0.9075 ± 0.0697 | 0.8945 ± 0.0865 | 0.9327 ± 0.0701 |

# 03 | Random Search

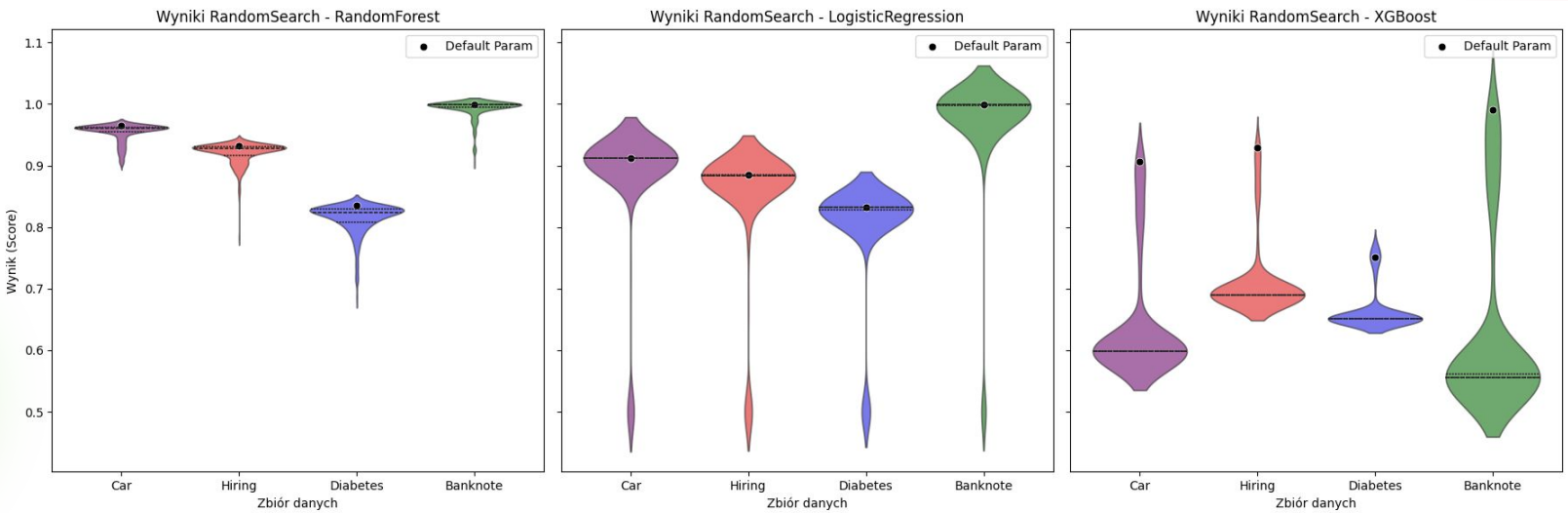


# 03 | Random Search



# *04*| Bayes Optimization

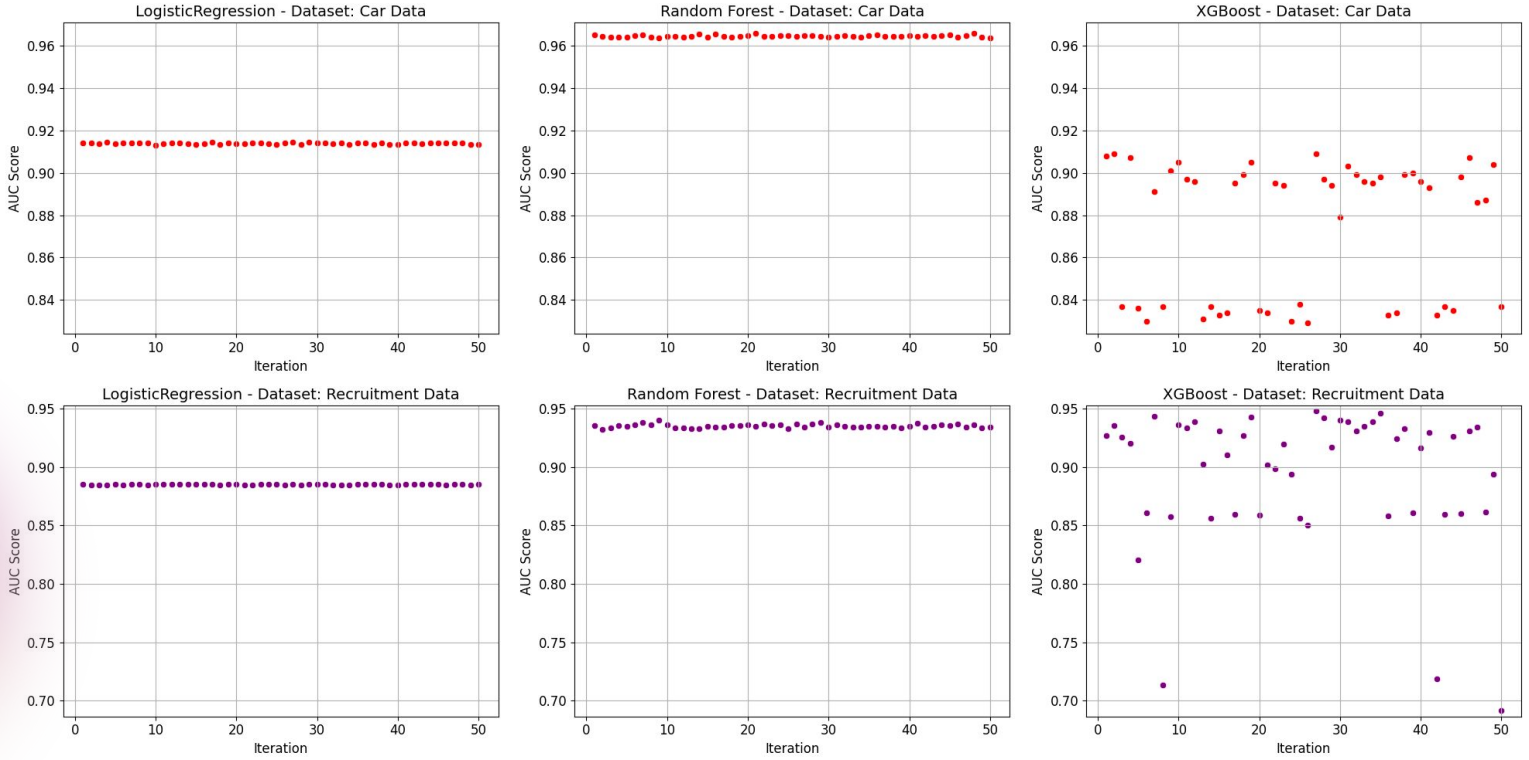
Kryterium optymalizacji: **AUC**

*Zakresy* hiperparametrów: takie same dla każdego zbioru danych, zróżnicowane w zależności od modelu

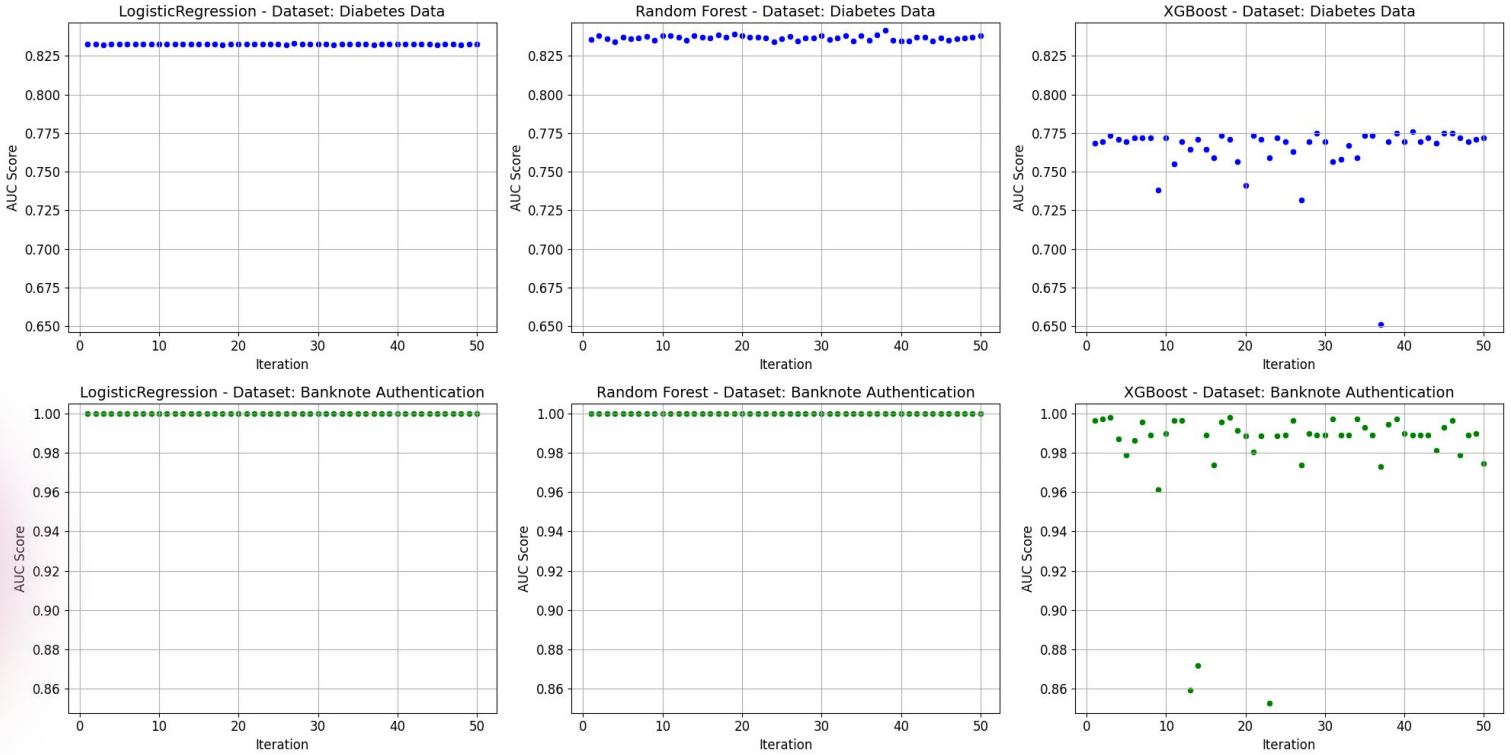
Liczba iteracji jednego wywołania: **30**

Liczba wywołań dla różnych punktów początkowych: **50**

# 04\ Bayes Optimization



# 04\ Bayes Optimization



# 05 | Wyniki – tunability

## Logistic Regression

| dataset | default* params (AUC) | best params random search (AUC) | best params bayes opt. (AUC) | **tunability** random search | **tunability** bayes opt. |
|---|---|---|---|---|---|
| car | 0.9126 | 0.9145 | 0.9145 | 0.0019 | 0.0019 |
| hiring | 0.8853 | 0.8859 | 0.8857 | 0.0006 | 0.0004 |
| diabetes | 0.8326 | 0.8327 | 0.8328 | 0.0001 | 0.0002 |
| banknote | 0.9995 | 0.9998 | 0.9998 | 0.0003 | 0.0003 |

# 05 | Wyniki – tunability

## Random Forest

| dataset | default* params (AUC) | best params random search (AUC) | best params bayes opt. (AUC) | **tunability** random search | **tunability** bayes opt. |
|---|---|---|---|---|---|
| car | 0.9644 | 0.9661 | 0.9658 | 0.0017 | 0.0014 |
| hiring | 0.9322 | 0.9380 | 0.9406 | 0.0058 | 0.0084 |
| diabetes | 0.8345 | 0.8368 | 0.8413 | 0.0023 | 0.0068 |
| banknote | 0.9997 | 0.9999 | 0.9999 | 0.0002 | 0.0002 |

# 05 | Wyniki – tunability

## XGBoost

| dataset | default* params (AUC) | best params random search (AUC) | best params bayes opt. (AUC) | **tunability** random search | **tunability** bayes opt. |
|---|---|---|---|---|---|
| car | 0.9070 | 0.9070 | 0.9090 | 0.0000 | 0.0020 |
| hiring | 0.9293 | 0.9380 | 0.9480 | 0.0087 | 0.0187 |
| diabetes | 0.7513 | 0.7734 | 0.7760 | 0.0221 | 0.0247 |
| banknote | 0.9905 | 0.9927 | 0.9978 | 0.0022 | 0.0073 |

# 05 | Wyniki +dodatek

## Logistic Regression

| dataset | default params from package authors | default* - default | best random - default | best bayes - default |
|---|---|---|---|---|
| car | 0.9133 | -0.0007 | **0.0012** | **0.0012** |
| hiring | 0.8872 | -0.0019 | -0.0013 | -0.0015 |
| diabetes | 0.8310 | **0.0016** | **0.0017** | **0.0018** |
| banknote | 0.9996 | -0.0001 | **0.0002** | **0.0002** |

# 05 | Wyniki +dodatek

## Random Forest

| dataset | default params from package authors | default* - default | best random - default | best bayes - default |
|---|---|---|---|---|
| car | 0.9583 | **0.0061** | **0.0078** | **0.0075** |
| hiring | 0.9336 | **0.0014** | **0.0044** | **0.0070** |
| diabetes | 0.8227 | **0.0118** | **0.0141** | **0.0186** |
| banknote | 0.9999 | -0.0002 | 0.0000 | 0.0000 |

# 05 | Wyniki +dodatek

## XGBoost

| dataset | default params from package authors | default* - default | best random - default | best bayes - default |
|---|---|---|---|---|
| car | 0.9599 | -0.0529 | -0.0529 | -0.0509 |
| hiring | 0.9402 | -0.0109 | -0.0022 | **0.0078** |
| diabetes | 0.7881 | -0.0368 | -0.0147 | -0.0121 |
| banknote | 0.9999 | -0.0094 | -0.0072 | -0.0021 |

# 06| Podsumowanie

**Optymalizacja:** Bayesian Optimization działa najlepiej dla złożonych modeli (np. XGBoost), ale różnice w stosunku do Random Search są niewielkie.

**Domyślne parametry:** Często wystarczają do osiągnięcia dobrych wyników, szczególnie w Random Forest i XGBoost.

**Praktyka:** Czasem warto skupić się na wyborze odpowiedniego modelu zamiast intensywnego tuningu hiperparametrów.

# Dziękujemy!

## Pytania?