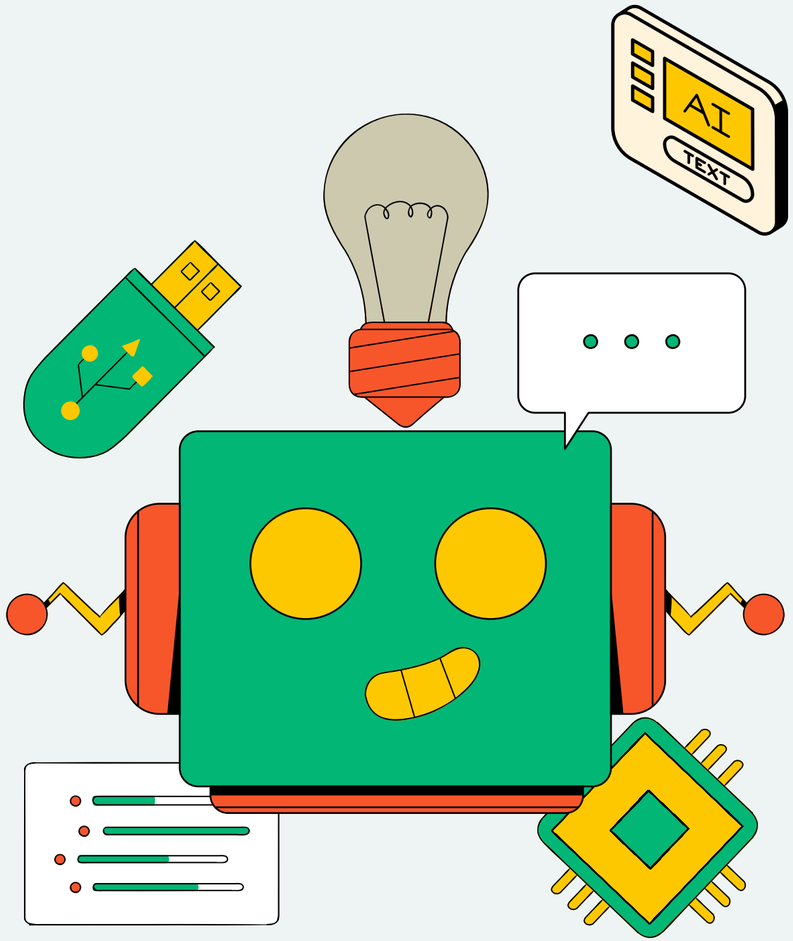


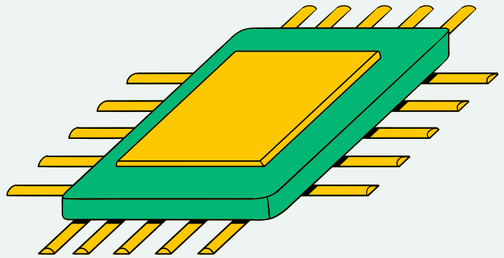
MINI AUTOML

AUTOML DO KLASYFIKACJI BINARNEJ

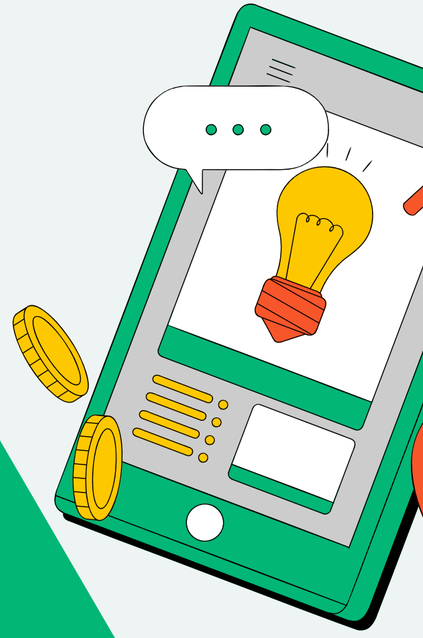
ALICJA PRZEŹDZIECKA

DARIA BARTKOWIAK

OLIWIA WÓJCICKA



STARTOWE PORTFOLIO



MODELE LINIOWE

SVC

BOOSTING

INSTANCE-BASED (KNN)

DRZEWIASTE

BAYES

WYBRANE KOMBINACJE

LOGISTIC REGRESSION

- **Standardowy model z solverem SAGA** (dobre dla dużych danych)
- **Klasyczny solver LBFGS** (stabilny)
- **Regularyzacja L1** (selekcja cech)
- **ElasticNet** (dane skorelowane)
- **RidgeClassifier**
- **LinearDiscriminantAnalysis (LDA):**
  - * "solver": "svd" (dużo cech)
  - * "solver": "lsqr", "shrinkage": "auto" (dane skorelowane)
- **Silna regularyzacja dla mniejszych zbiorów** (małe, trudne dane)

DRZEWA DECYZYJNE

- **Bardzo płytkie (Decision Stump)**
- **Bezpieczny wybór (Średnia złożoność)**
- **Głębokie drzewo (Wysoka złożoność)**
- **Bez ograniczeń (Eksploracja)**

RANDOM FOREST

- **Szybki, prosty, płytki**
- **Pełne drzewo (Mocny model)**
- **Głębokie z ograniczeniami (Bezpieczniejsze)**
- **Eksploracja entropii**

WYBRANE KOMBINACJE

HGBOOST

- Płytki, szybko uczący się
- Głębszy, umiarkowana szybkość
- Średni model (Baseline)
- Slow Learner (Bardzo wolna nauka)
- Głęboki z ograniczeniami
- Regularyzacja (L1/L2)

CATBOOST

- Standardowy
- Wolniejszy z regularyzacją
- Mocny model (Intensive)
- Długodystansowiec
- Łagodniejszy wariant
- Płytki (Small)
- Deep Slow (Eksperymentalny)

KNN

- Średnie K, wagi odległościowe
- Duże K
- Domyślny
- Wysokowymiarowy

WYBRANE KOMBINACJE

SVC

- Bazowy o średnich parametrach
- Standardowy RBF
- Liniowy (Soft Margin)
- Wielomianowy
- Hard Margin (Rygorystyczny)

EXTRA TREES

- Wariant zrównoważony
- Wariant głęboki

INNE

- sklearn.naive_bayes.GaussianNB
- sklearn.naive_bayes.BernoulliNB
- sklearn.ensemble.AdaBoostClassifier
- sklearn.ensemble.GradientBoostingClassifier
- sklearn.ensemble.HistGradientBoostingClassifier

WYBÓR FINALNEGO PORTFOLIO

- **Dane:** zestaw zadań binarnych z OpenML (różne typy danych)
- Każdy model testowany w identycznym pipeline:
  - preprocessing (imputacja + Yeo–Johnson + OneHot)
  - metryka: AUC
- **Scoring stabilności** (średnia jakość + kara za zmienność):
  - **$\underline{\text{Score} = \text{Mean\_AUC} - \lambda \cdot \text{VolatilityScore}}$**
- Strategia wyboru:
  a. **Diversity**: najlepszy model z każdej rodziny
  b. **Performance**: uzupełnienie listy do 30 najlepszymi score
  c. **Priorytetyzacja**: ustawienie kolejności pod ograniczenia czasu

ENSEMBLE

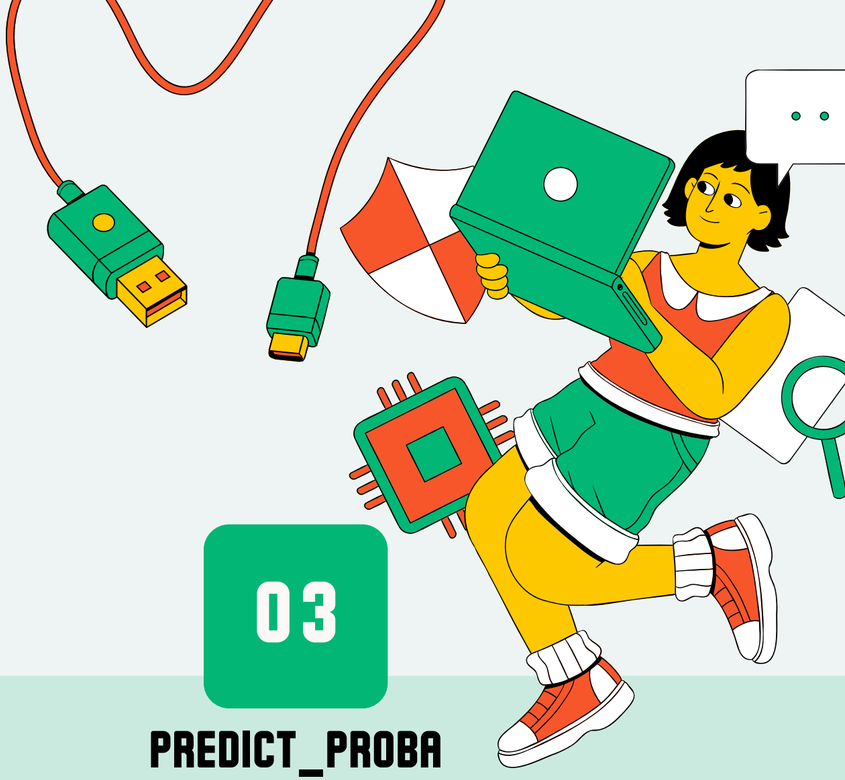
Kandydaci do ensemble:

- **wybieramy top_k modeli (domyślnie 5)**
- **priorytet: różne rodziny algorytmów → mniejsza korelacja błędów**

Mechanizm:

- soft voting = uśrednianie prawdopodobieństw klas
- dobieramy optymalny próg decyzyjny (0.2–0.8)
- **metryka**: Balanced Accuracy

PAKIET MINI AUTOML



01 FIT

- buduje preprocessing (ColumnTransformer)
- przeprowadza turniej modeli z portfolio
- dobiera optymalny próg decyzyjny (0.2–0.8)
- opcjonalnie buduje ensemble soft-voting
- wybiera najlepszy wariant
- wykonuje retraining na pełnym zbiorze
- zapisuje: **final_model, final_threshold, leaderboard**

02 PREDICT

- preprocessing → predykcja
- jeśli dostępne predict_proba: stosuje threshold
- zwraca etykiety {0,1}

03 PREDICT_PROBA

- zwraca prawdopodobieństwa klas
- fallback: jeśli brak predict_proba, zwraca macierz z predykcji 0/1

WYBRANIE NAJLEPSZEGO MODELU - FIT

PREPROCESSING

- **numeryczne:** median imputer + Yeo–Johnson
- **kategoryczne:** missing + One-Hot Encoding

KONTROLA KOSZTU SELEKCJI

jeśli **n_samples > max_rows_limit →** losowanie podzbioru do selekcji

OGRANICZENIE CZASOWE

- **parametr:** total_time_limit (minuty)
- Budżet dzielimy na:
  - Search budget (turniej modeli)
  - Reserve time na finalny trening zwycięzcy
- Rezerwa czasu:
  - reserve_time = max(1 min, 25% limitu)
  - dla limitu < 5 min → reserve_time = 0.5 min
- Warunek przerwania turnieju:
- Jeśli przekroczono search_budget_sec → **STOP selekcji**
- **W praktyce**: testujemy tyle modeli, ile się zmieści w czasie

TURNIEJ MODELI

- **split 80/20 stratified hold-out**
- każdy model trenowany na train i oceniany na val
- **metryka**: Balanced Accuracy
- tuning progu
- opcjonalny ensemble

WYBÓR FINALNY

- porównanie **BalAcc(single) vs BalAcc(ensemble)**
- retraining zwycięzcy na pełnym zbiorze

LEADERBOARD | OUTPUT

```
                                MINI AUTOML
================================================================================
[DATA] Loading data from: 'example_data'
[CONF] Configuration:     'models.json'
[INFO] Dataset loaded:    3481 samples, 16 features
[INFO] Class Balance:     Class False: 55.8% | Class True: 44.2%
--------------------------------------------------------------------------------
[TRAIN] Starting search (Time Budget: 5 min, Top-K: 5)...
--- TIME BUDGET: 5 min (Search: 3.8 min) ---
Preprocessing data...
Starting model tournament...
Model 1/30: lda1 -> BalAcc: 0.5573
Model 2/30: svm2 -> BalAcc: 0.5586
Model 3/30: random_forest3 -> BalAcc: 0.5499
Model 4/30: xgboost5 -> BalAcc: 0.5615
Model 5/30: ridge_clf -> BalAcc: 0.5526
Model 6/30: lda2 -> BalAcc: 0.5553
Model 7/30: log_reg2 -> BalAcc: 0.5553
Model 8/30: log_reg1 -> BalAcc: 0.5526
Model 9/30: xgboost4 -> BalAcc: 0.5543
Model 10/30: log_reg3 -> BalAcc: 0.5567
Model 11/30: xgboost1 -> BalAcc: 0.5536
Model 12/30: xgboost2 -> BalAcc: 0.5692
Model 13/30: xgboost6 -> BalAcc: 0.5731
...
Model 23/30: log_reg4 -> BalAcc: 0.5589
Model 24/30: ada_boost1 -> BalAcc: 0.5359
Model 25/30: ada_boost2 -> BalAcc: 0.5420
Model 26/30: log_reg5 -> BalAcc: 0.5645
```

```
Best Single Model: svm4 (BalAcc: 0.6051)
/Library/Frameworks/Python.framework/Versions/3.13/lib/python3.13/site-packages/sklear
  warnings.warn(
Ensemble score: 0.5783
>>> Selected: SINGLE (svm4)
Retraining winning model on full data...
Done! Total time: 0.28 min.
[DONE] Process completed in: 16.80 s

================================================================================
                          INTERNAL LEADERBOARD (TOP 10)
================================================================================
         name             family     score  threshold
                 svm4          svm 0.6051    0.45
                 svm1          svm 0.5823    0.50
          extra_trees2        other 0.5782    0.50
              xgboost6      xgboost 0.5731    0.55
                  knn2          knn 0.5708    0.40
              xgboost2      xgboost 0.5692    0.60
              log_reg5       linear 0.5645    0.55
              xgboost3      xgboost 0.5644    0.45
hist_gradient_boosting  gbm_sklearn 0.5629    0.45
        random_forest4 randomforest 0.5622    0.50
--------------------------------------------------------------------------------

############################################################
                    FINAL TEST SET RESULTS
############################################################
...
 -> Type: SVC
 -> Cutoff Threshold: 0.4500
############################################################
```

DZIĘKUJEMY ZA UWAGĘ :D