

MOBILE PHONES

PRICE RANGE

CLASSIFICATION

Martyna Leśniak
Urszula Szczesna

DATA

Dane pochodzą z kaggle : <https://www.kaggle.com/datasets/ahmedghonem01/phones-price-classification>

20 kolumn danych numerycznych z parametrami telefonów

target to price_range o wartości z zakresu 0-3

Dane zostały podzielone na dane treningowe, testowe i treningowe (80/10/10)

```
Data columns (total 21 columns):
#   Column              Non-Null Count  Dtype
---  -
0   battery_power        1600 non-null   int64
1   blue                  1600 non-null   int64
2   clock_speed          1600 non-null   float64
3   dual_sim             1600 non-null   int64
4   fc                   1600 non-null   int64
5   four_g               1600 non-null   int64
6   int_memory           1600 non-null   int64
7   m_dep                1600 non-null   float64
8   mobile_wt            1600 non-null   int64
9   n_cores              1600 non-null   int64
10  pc                    1600 non-null   int64
11  px_height             1600 non-null   int64
12  px_width              1600 non-null   int64
13  ram                   1600 non-null   int64
14  sc_h                  1600 non-null   int64
15  sc_w                  1600 non-null   int64
16  talk_time             1600 non-null   int64
17  three_g               1600 non-null   int64
18  touch_screen          1600 non-null   int64
19  wifi                  1600 non-null   int64
20  price_range           1600 non-null   int64
dtypes: float64(2), int64(19)
```

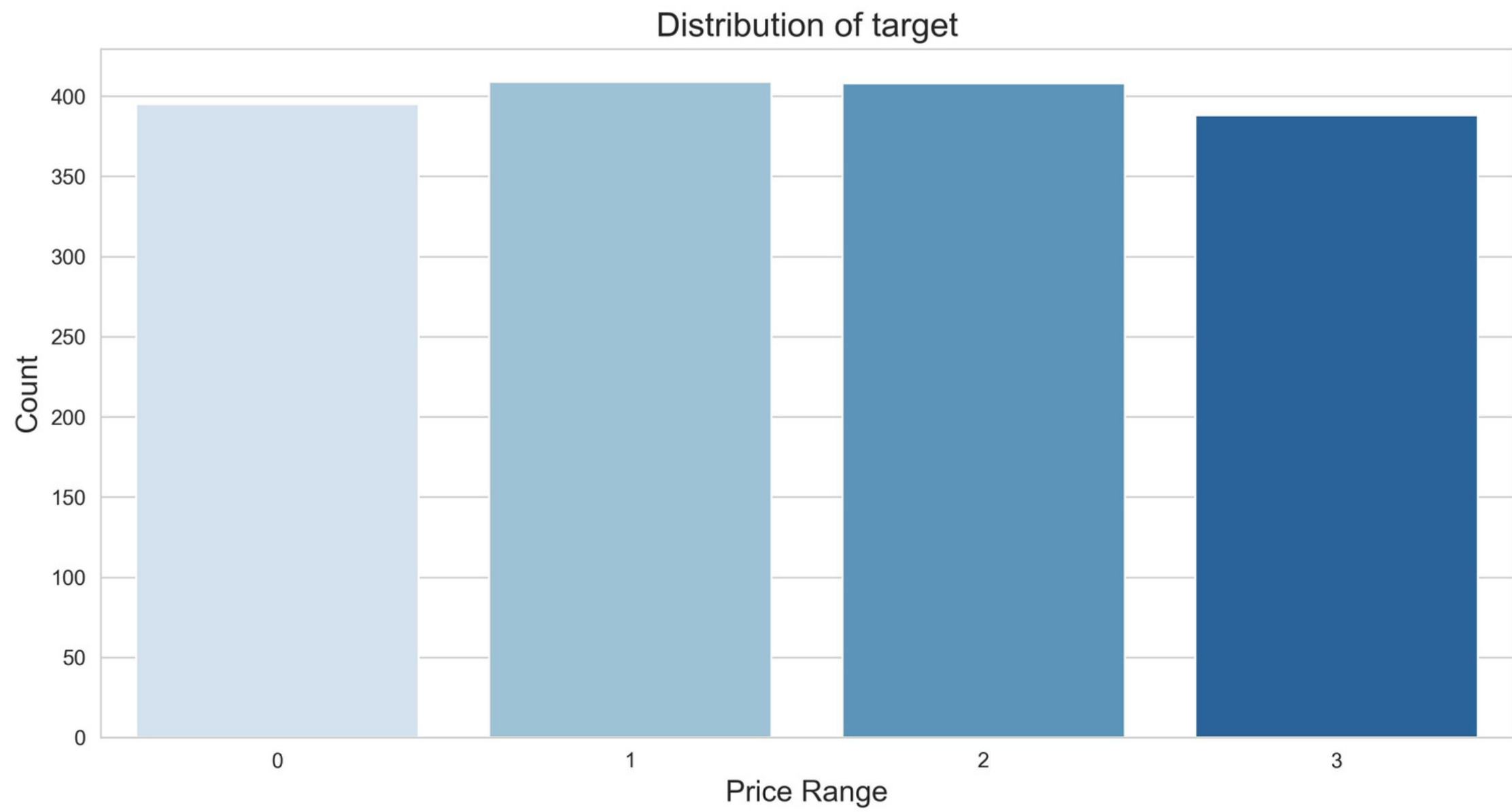
EDA

Rozkład klas jest równomierny -> główna metryka accuracy

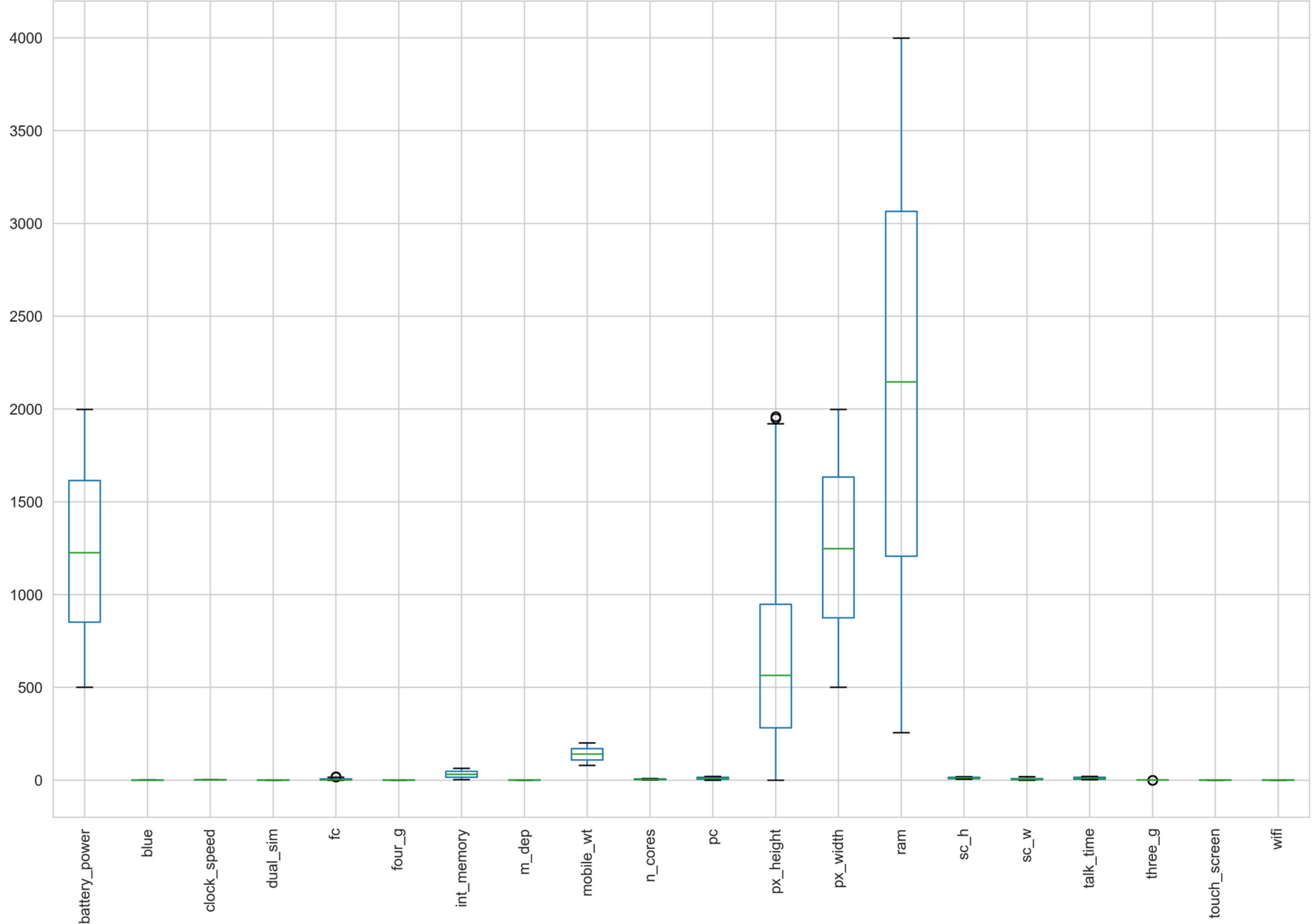
rozróżniał liczbowy pomiędzy różnymi parametrami jest duży

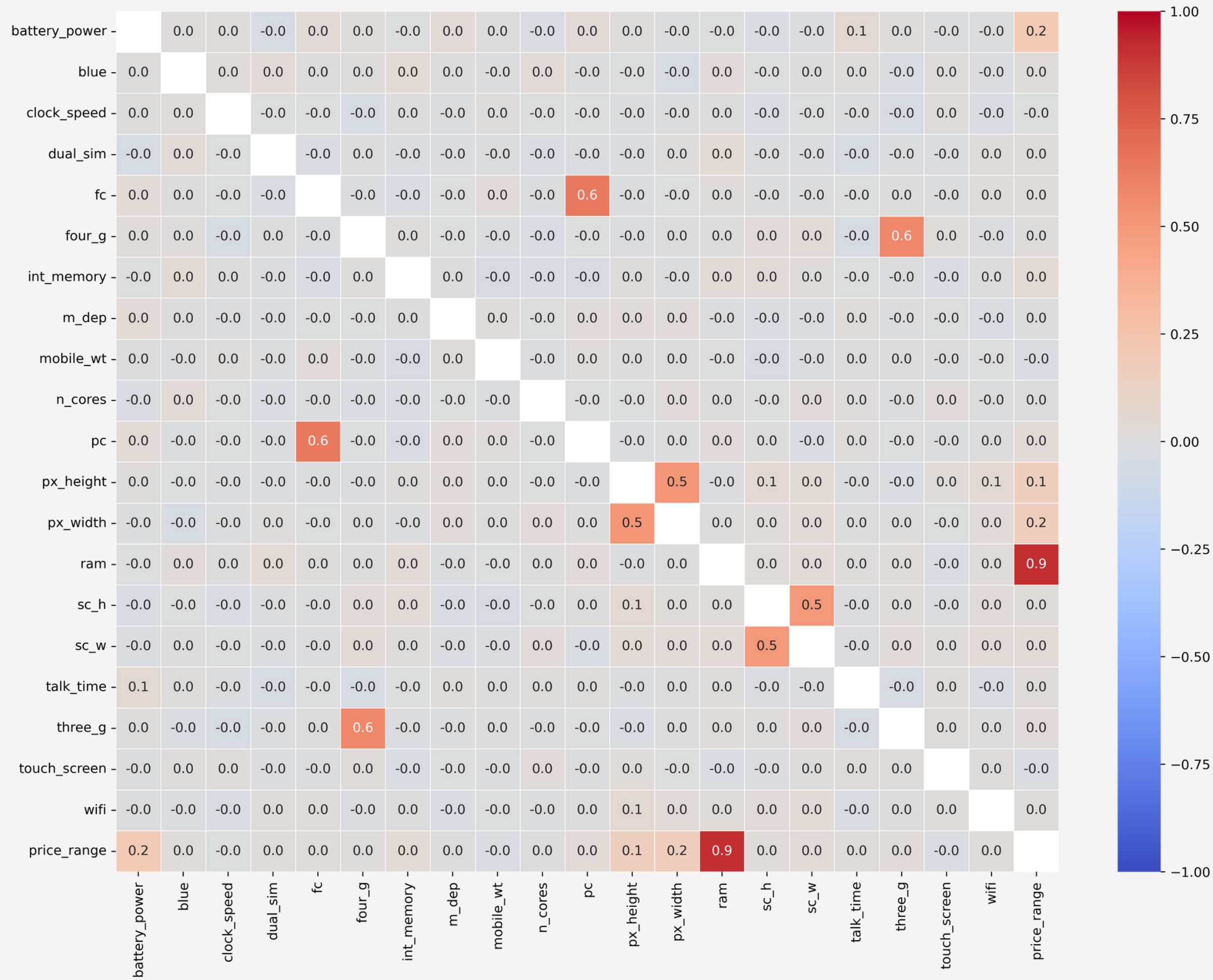
korelacja pomiędzy większością parametrów jest znikoma

Parametry o największej korelacji ze zmienną celu: ram, battery power, i pixel resolution



Distribution of all variables



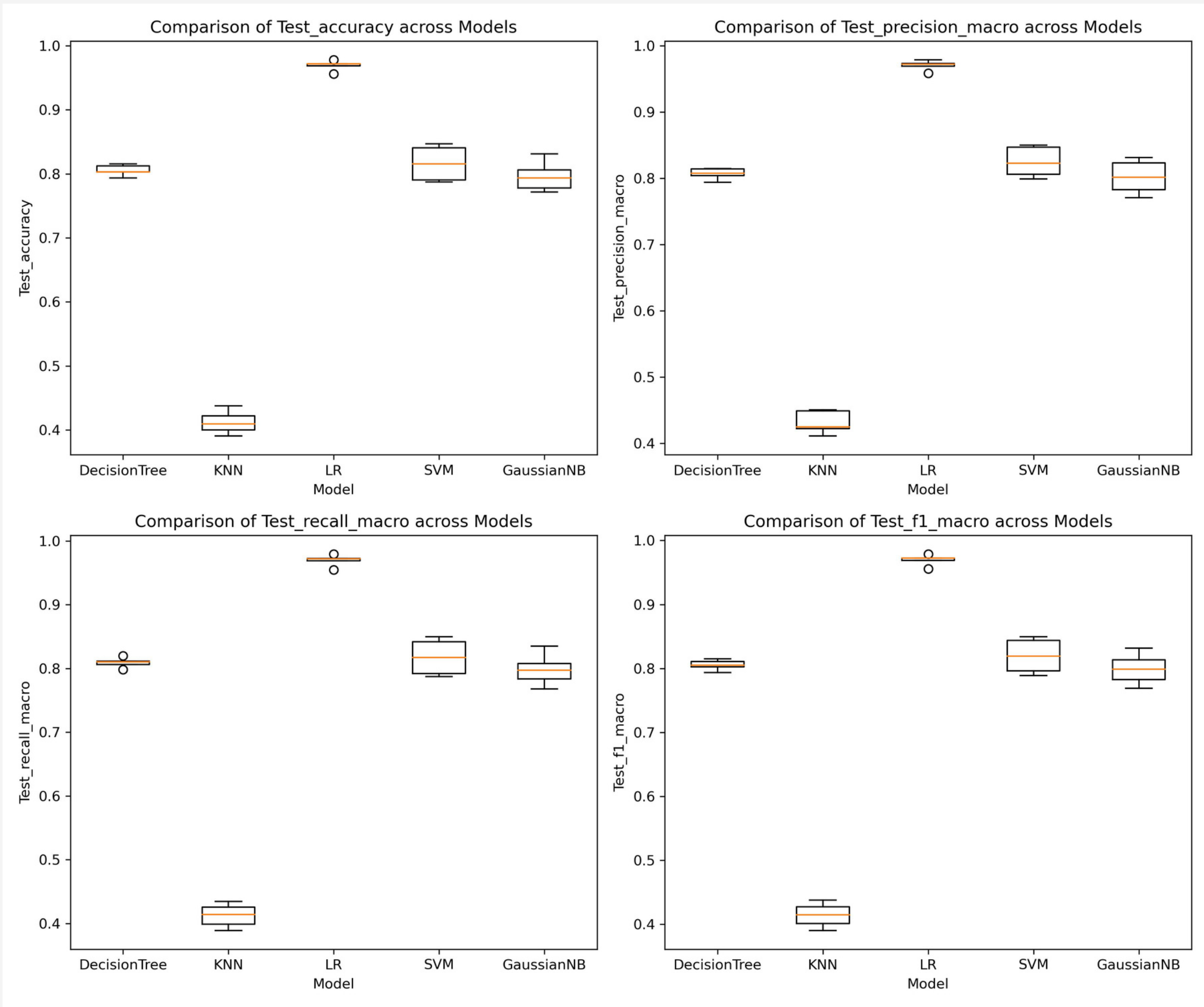


FEATURE ENGINEERING AND SIMPLE MODELS

Ze względu na duży rozrzut liczbowy parametrów użyliśmy skalowania (MinMax Scaler)

Pierwsze użyte modele:

- Decision Tree
- K- Nearest Neighbors
- Logistic Regression
- Supported Vector Machine
- Gaussian Naive Bayes



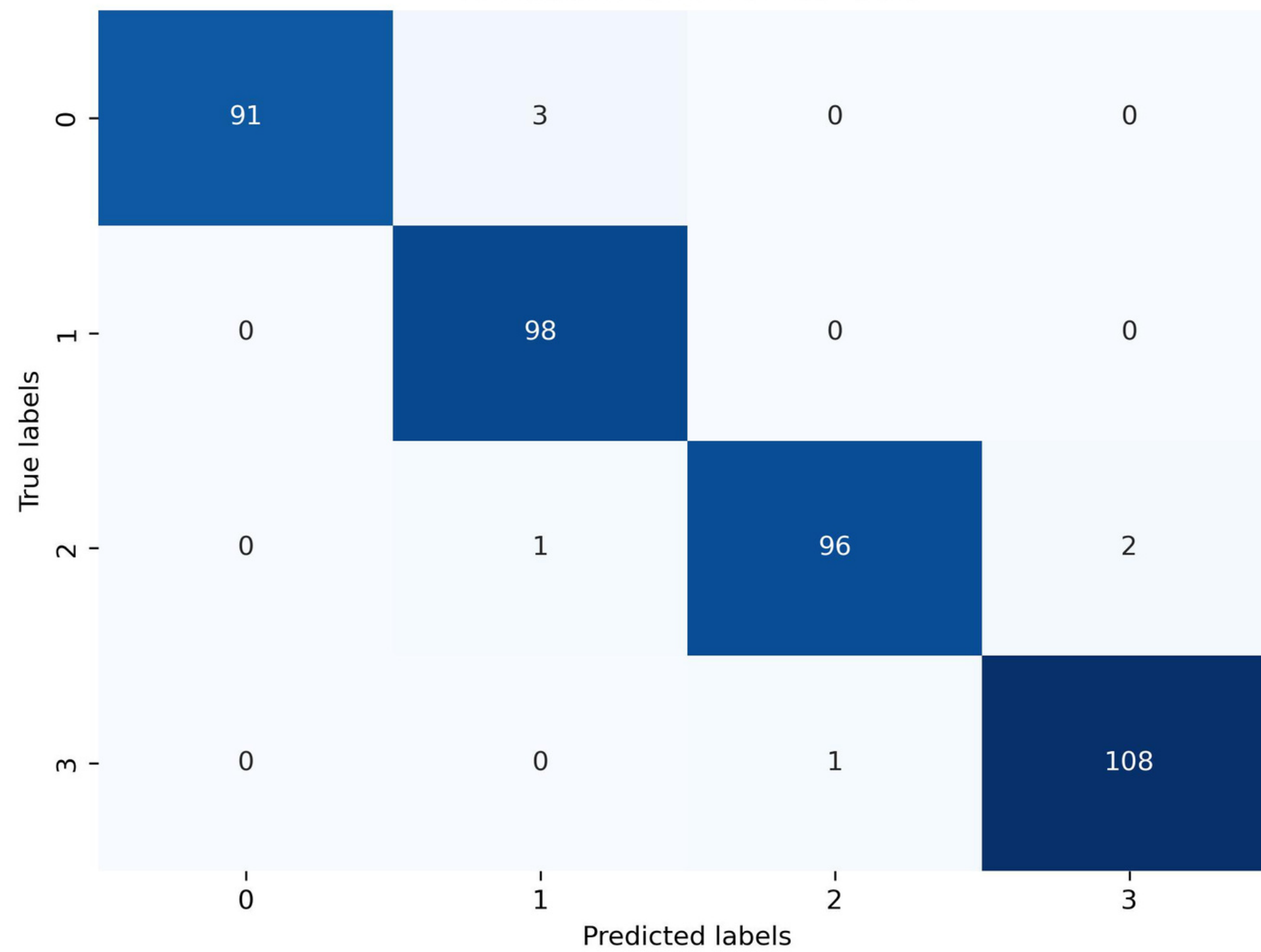
LOGISTIC REGRESSION

Ze względu na bardzo dobry wynik tego modelu postanowiliśmy go dalej ewaluować.

Za pomocą Grid Search znaleźliśmy najlepsze hiperparametry.

| Classification Report: | | | | |
|------------------------|-----------|--------|----------|---------|
| | precision | recall | f1-score | support |
| 0 | 1.00 | 0.97 | 0.98 | 94 |
| 1 | 0.96 | 1.00 | 0.98 | 98 |
| 2 | 0.99 | 0.97 | 0.98 | 99 |
| 3 | 0.98 | 0.99 | 0.99 | 109 |
| accuracy | | | 0.98 | 400 |
| macro avg | 0.98 | 0.98 | 0.98 | 400 |
| weighted avg | 0.98 | 0.98 | 0.98 | 400 |

Confusion Matrix for test data



FEATURE SELECTION

Zastanowiło nas czemu model KNN ma tak słaby wynik.

Za pomocą SelectKBest wybrałyśmy kombinację kolumn z najwyższym accuracy.

Wybrane kolumny:

- ram
- battery power
- px width
- px height

STACKED CLASSIFIER

Wybrałyśmy proste modele, które dawały najlepsze wyniki.

Stacked Classifier składa się z:

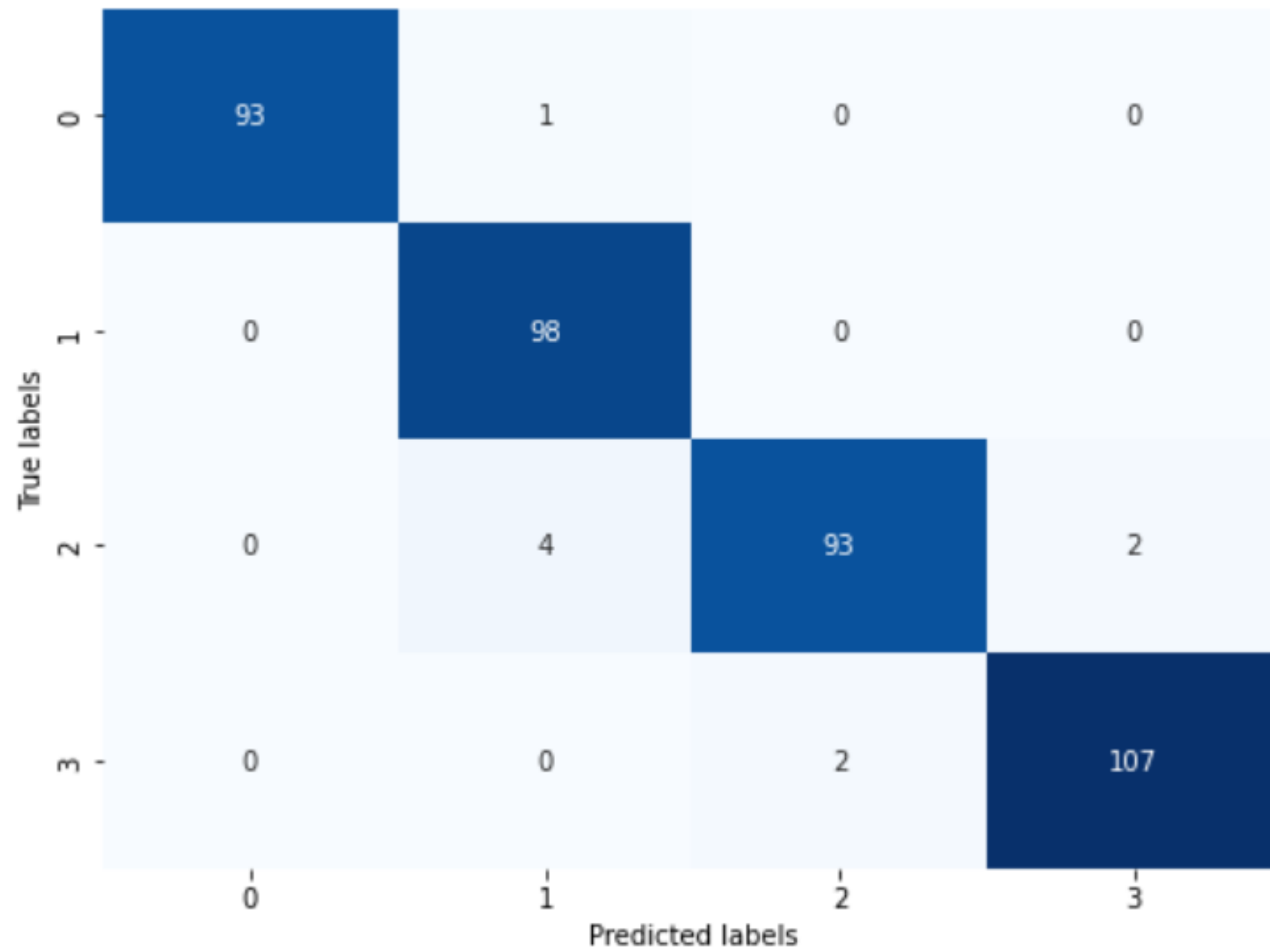
- K- Nearest Neighbors
- SVM
- Decision Tree
- Logistic Regression – finalny estymator

```
Classification Report:
              precision    recall  f1-score   support

     0           1.00        0.99        0.99         94
     1           0.95        1.00        0.98         98
     2           0.98        0.94        0.96         99
     3           0.98        0.98        0.98        109

 accuracy              0.98         400
 macro avg           0.98         0.98        0.98         400
 weighted avg           0.98         0.98        0.98         400
```

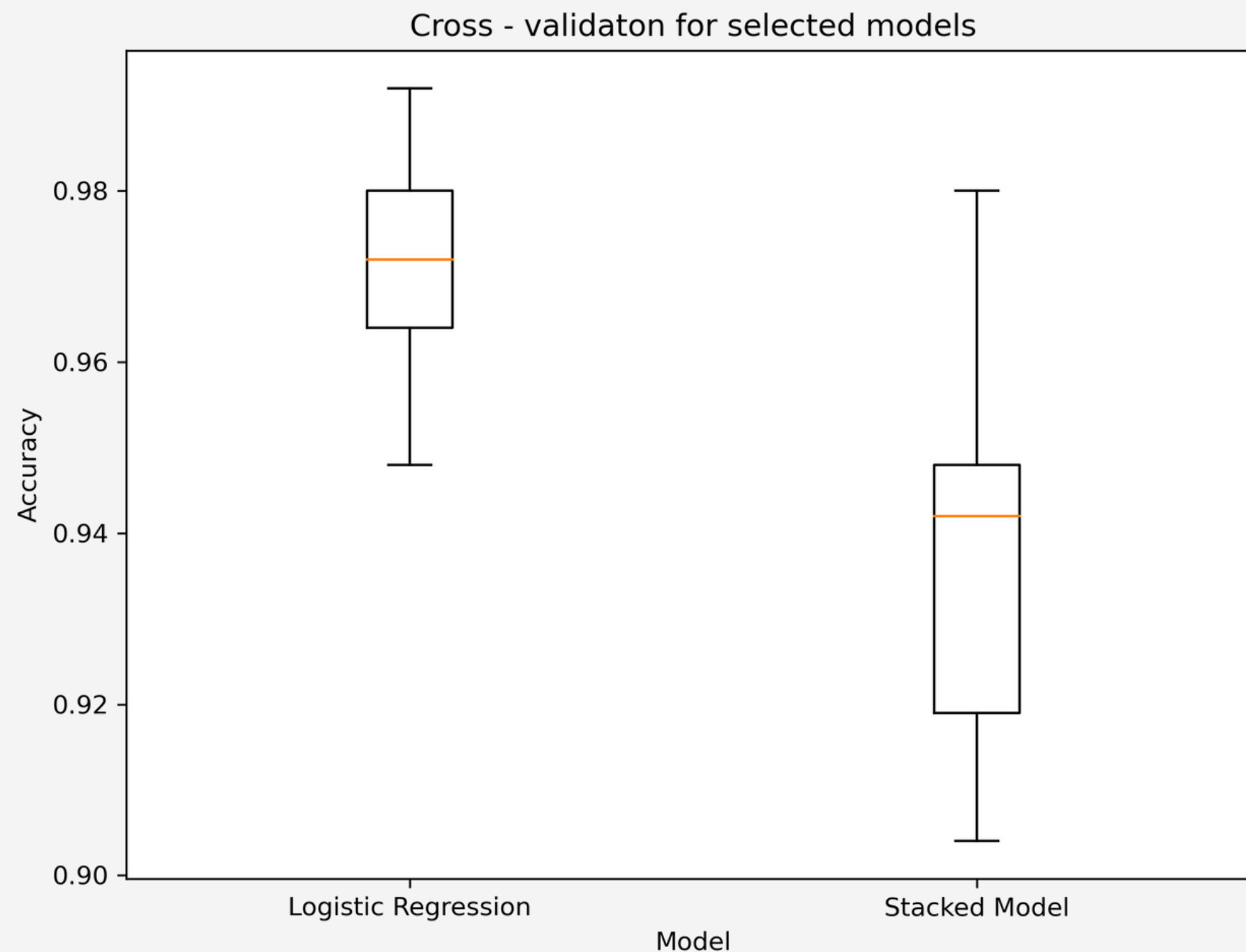
Confusion Matrix



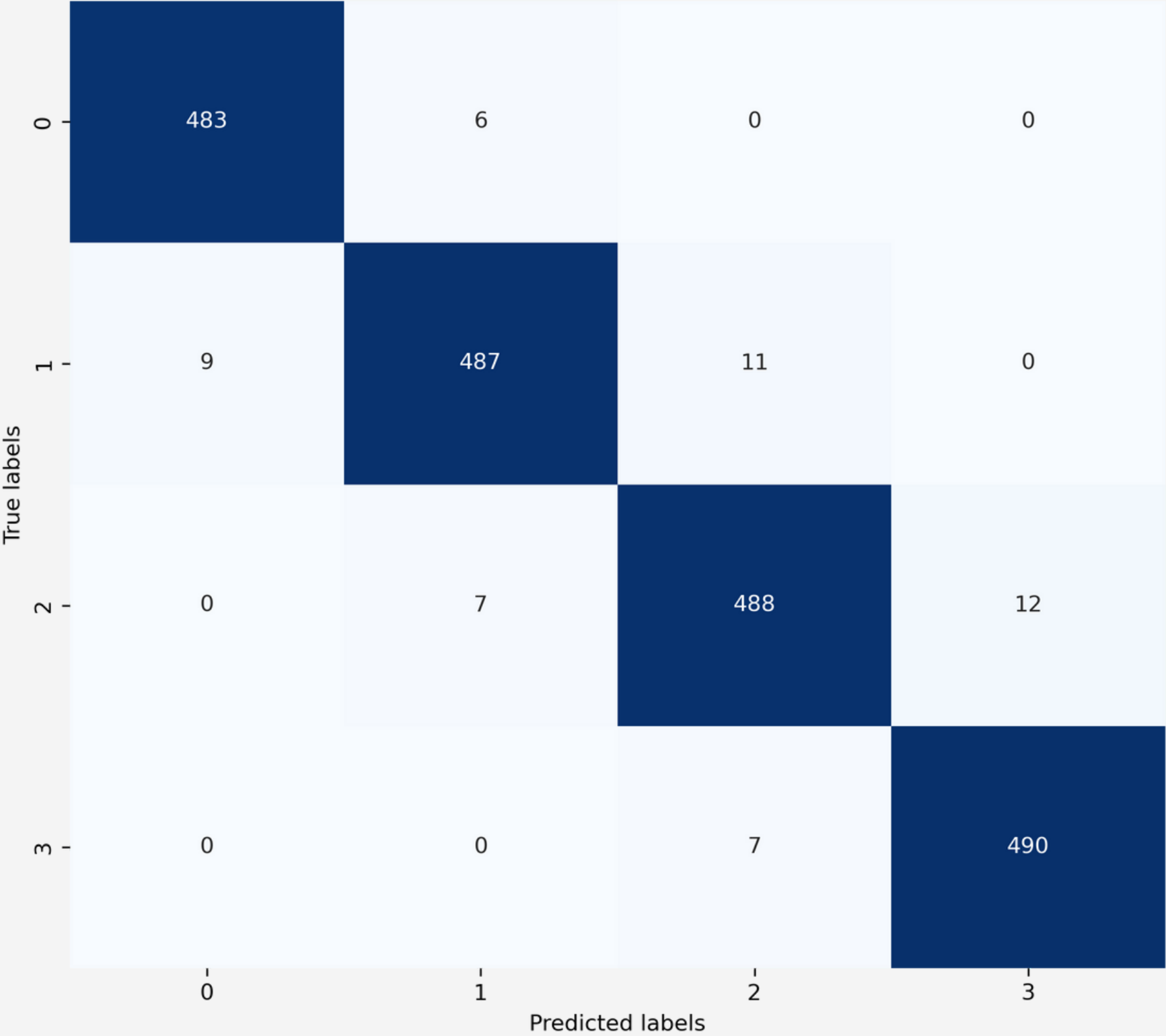
MODELS COMPARISON

Walidacja krzyżowa

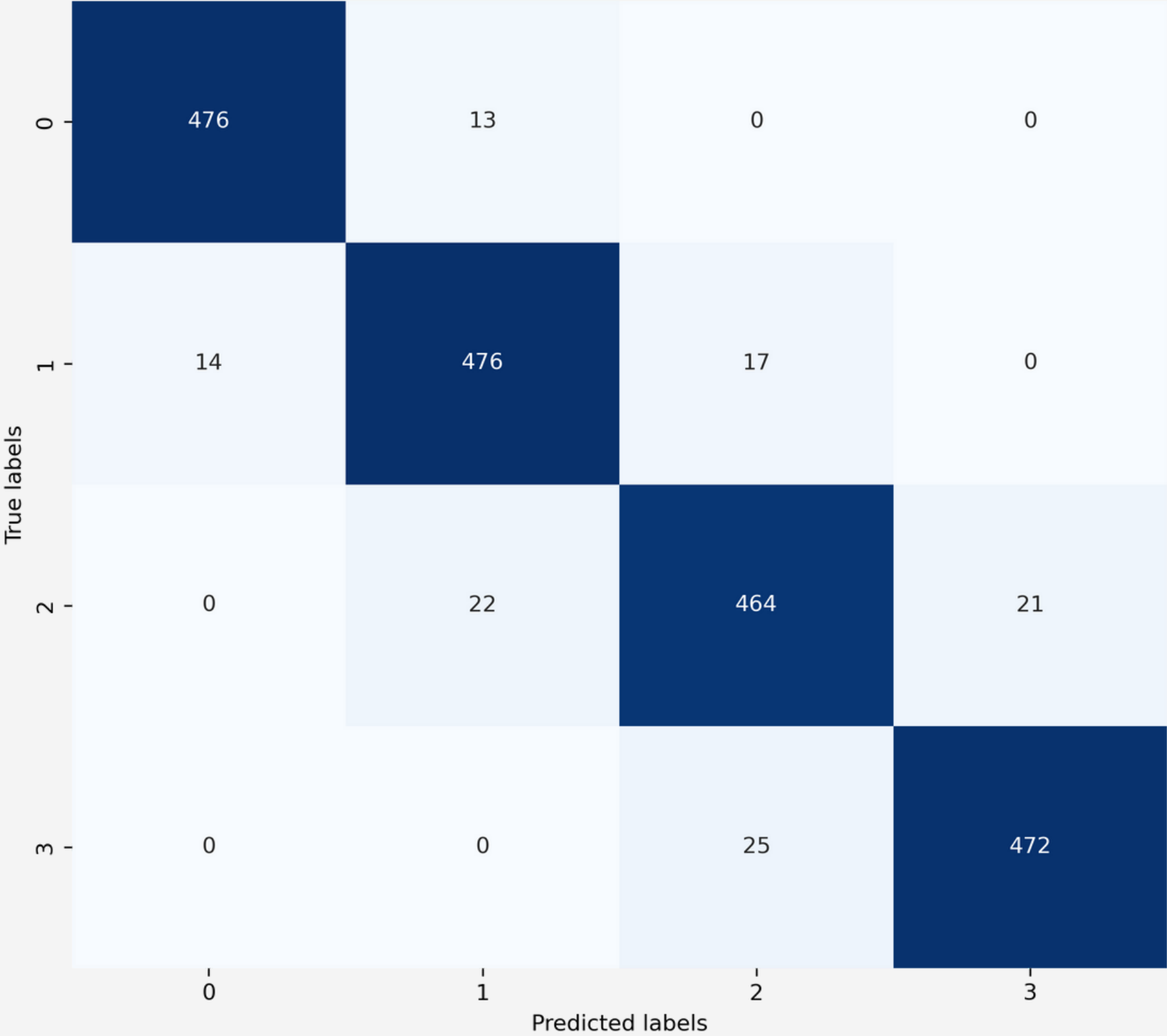
Średnia z accuracy: **0.9715 Logistic Regression** , **0.9385 Stacked Classifier**



Confusion matrix for Logistic Regression



Confusion matrix for Stacked Classifier



XAI

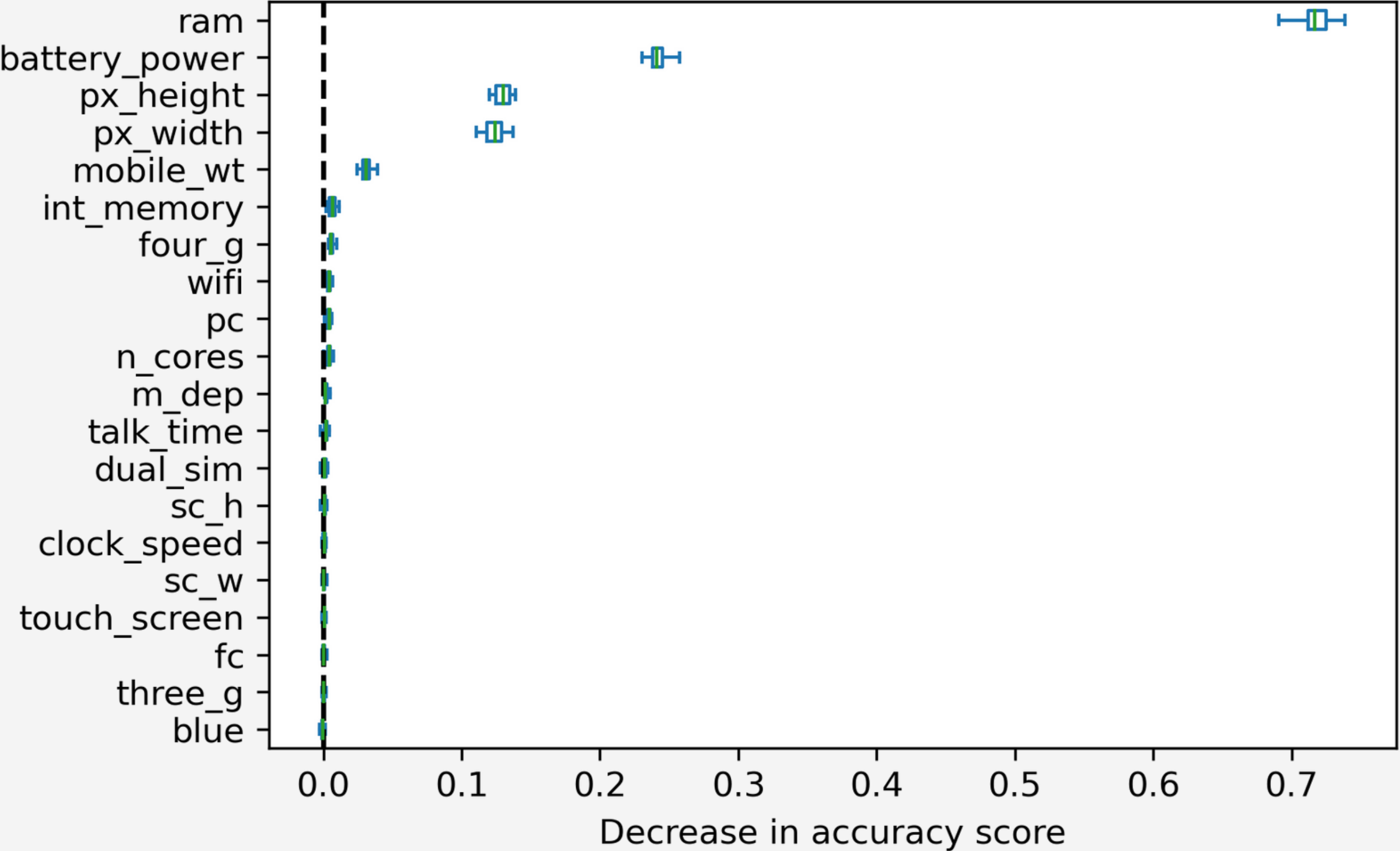
Jak model podejmuje decyzje?

Które zmienne mają największy wpływ?

Wykorzystane metody:

- `model_parts`
- `permutationo_importance`

Permutation Importances



Permutation Importances

