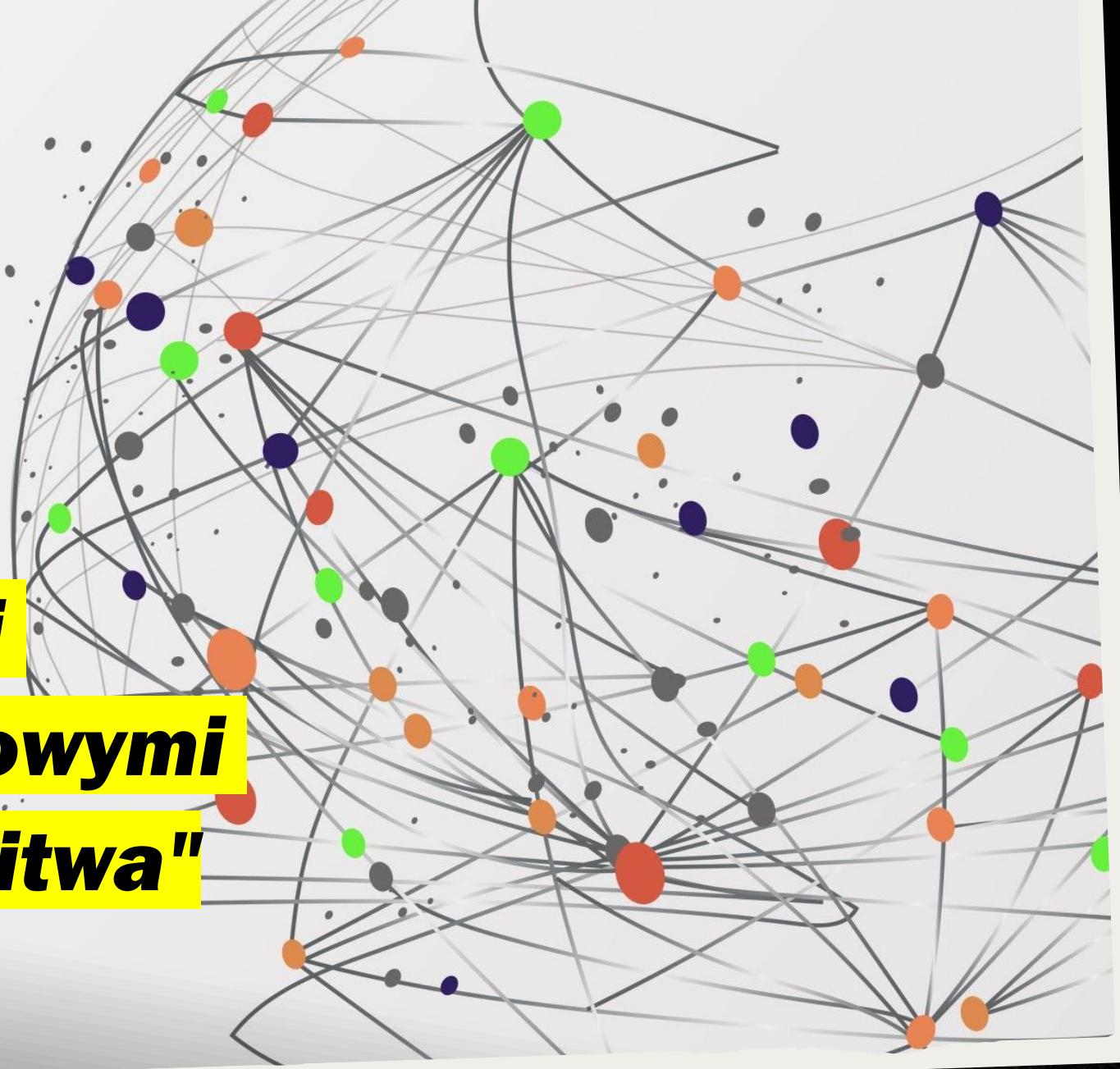
Badania klasyfikatorami i sieciami neuronowymi autorskiej gry "Bitwa"



Marcin Wojciechowski

Repo + bibliografia:

Repozytorium: https://gitlab.com/mwojciechowski653/io-projekt-1

PCA: https://builtin.com/machine-learning/pca-in-python

Drzewo decyzyjne: https://scikit-learn.org/stable/modules/tree.html

KNN: https://stackabuse.com/k-nearest-neighbors-algorithm-in-python-and-scikit-learn/

Bayes: https://www.datacamp.com/tutorial/naive-bayes-scikit-learn

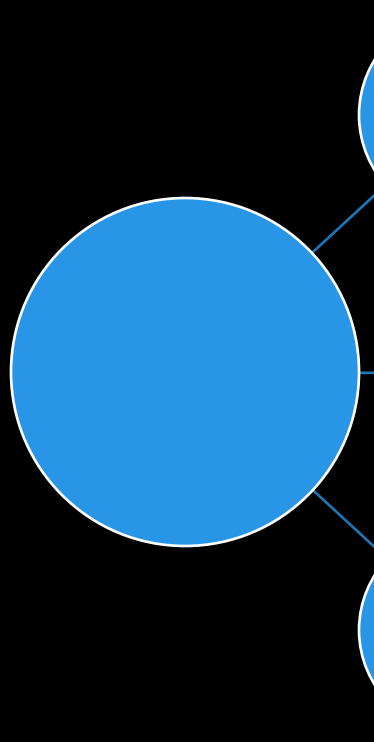
Sieci Neuronowe:

https://scikit-learn.org/stable/modules/generated/sklearn.neural_network.MLPClassifier.html

https://www.pluralsight.com/resources/blog/guides/machine-learning-neural-networks-scikit-learn

Wykresy: https://matplotlib.org/stable/tutorials/pyplot.html

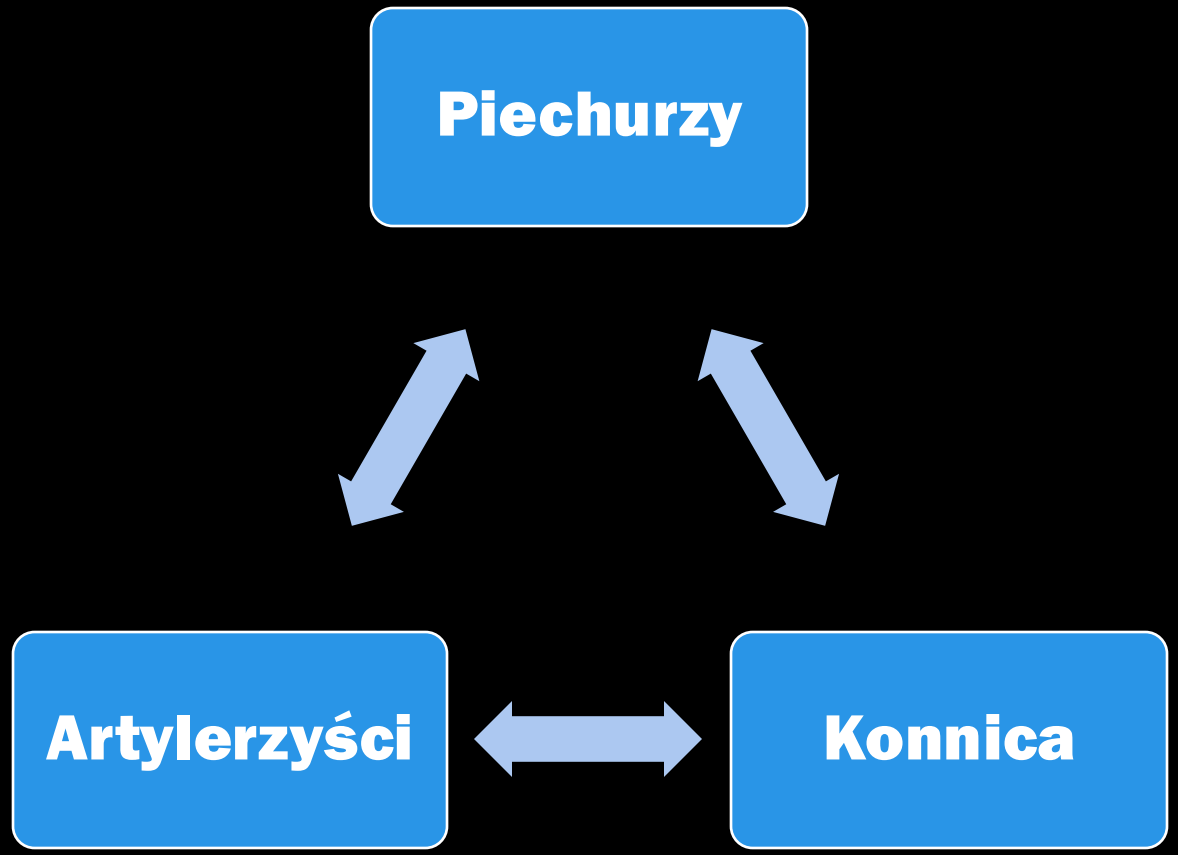
Założenia gry



- Ilość
- typ

- ilość
- typ

- ilość
- typ



Baza danych

Autorska, rekordy tworzone są przez własny generator dzielący te same funkcje, co gra dla użytkownika**, train_size = 0.7**

13 kolumn, po 6 na jedną armię i **końcowa**, czy bitwa została wygrana (1, przegrana 0) przez armię numer 1 (naszą)

700pkt vs 1000pkt, dlatego założyłem, że w bazie ma być minimum 20% wygranych

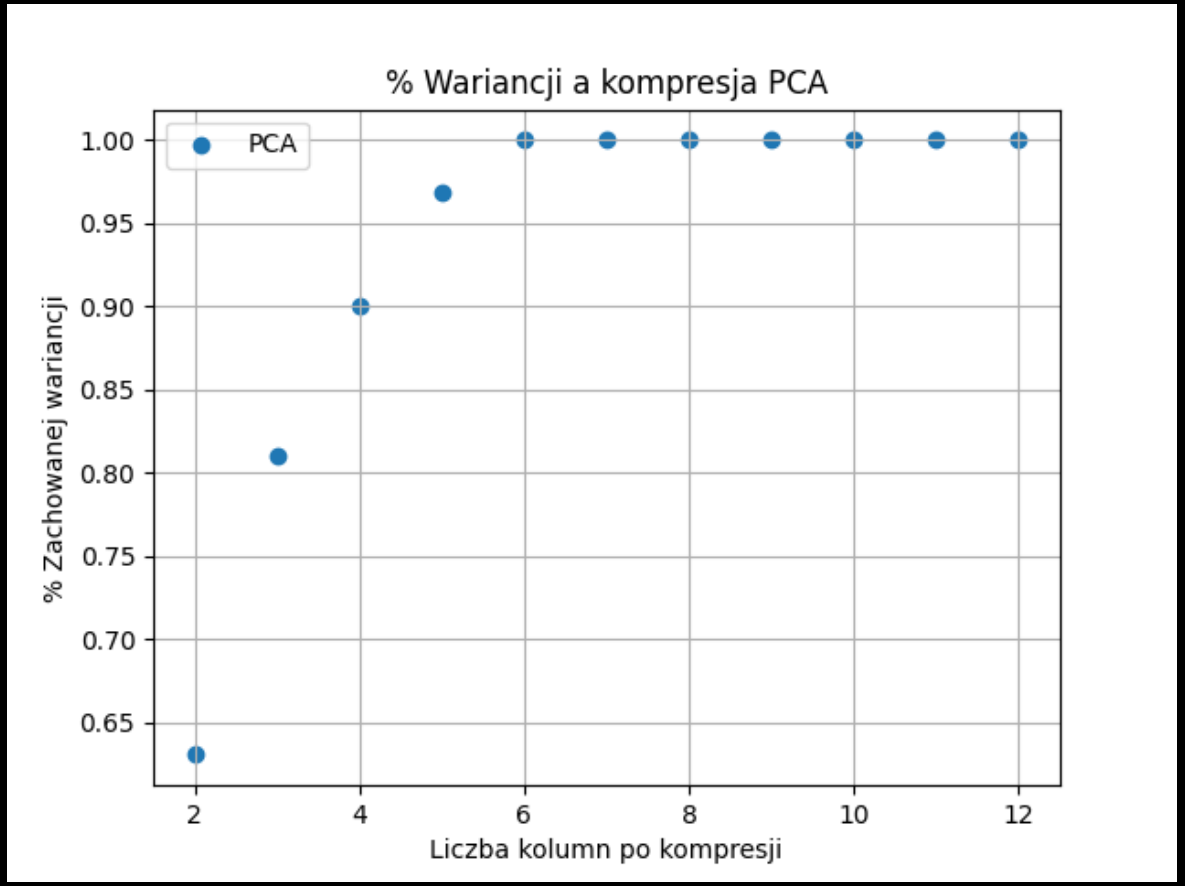
W bazie do badań znalazło się 2000 rekordów, 533 wygrane i 1467 przegranych, a więc było 26,65% wygranych - założenie spełnione

PCA

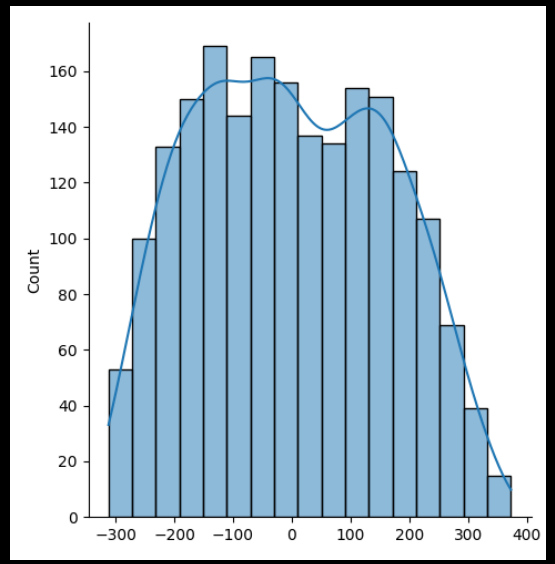
Dla redukcji do:

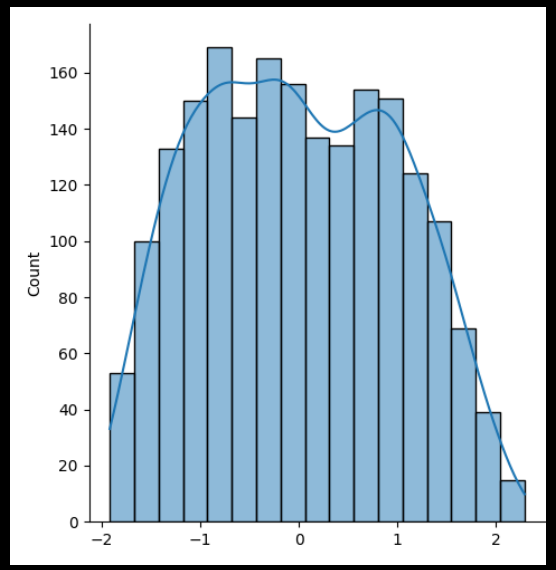
-6 kolumn zachowana wariancja wyniosła: 0.9999355,

-5 kolumn: 0.9688449.



Normalizacja





Dla redukcji do 5 kolumn wartości w bazie znajdowały się w przedziale od -311.87 do 394.12.

Do tego trafiają się elementy około 0 i w wielu innych miejscach, dlatego zdecydowałem się na badanie standaryzacji (z-score).

Po przeprowadzeniu standaryzacji średnia dla pierwszej kolumny wyniosła: 6.4659e-15, a odchylenie standardowe: 162.72.

Drzewo decyzyjne

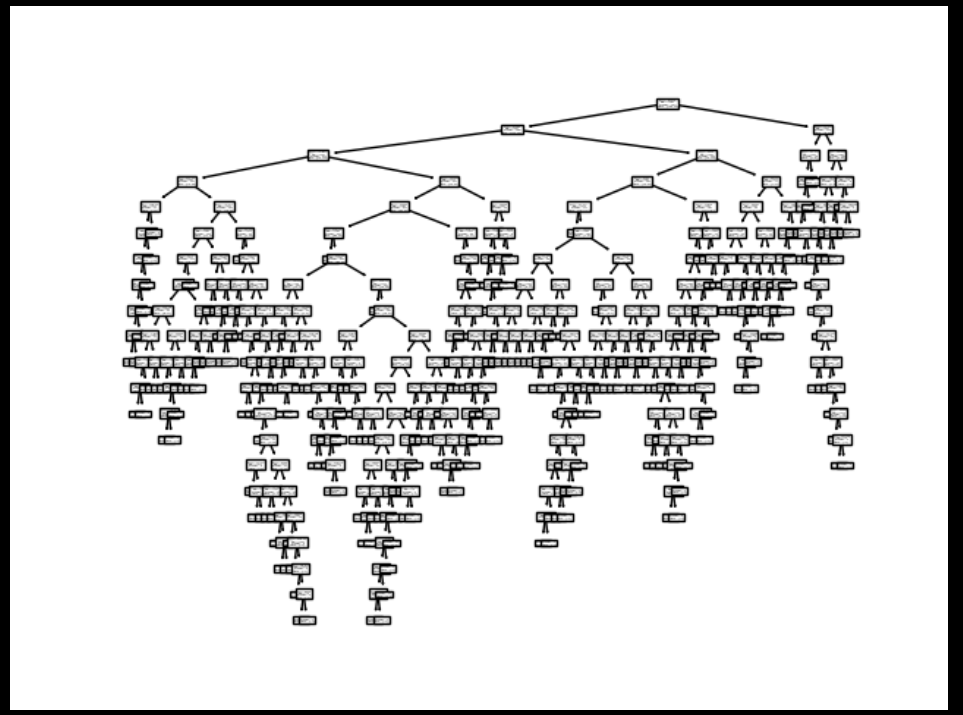
Macierz błędów dla przedstawionego drzewa:

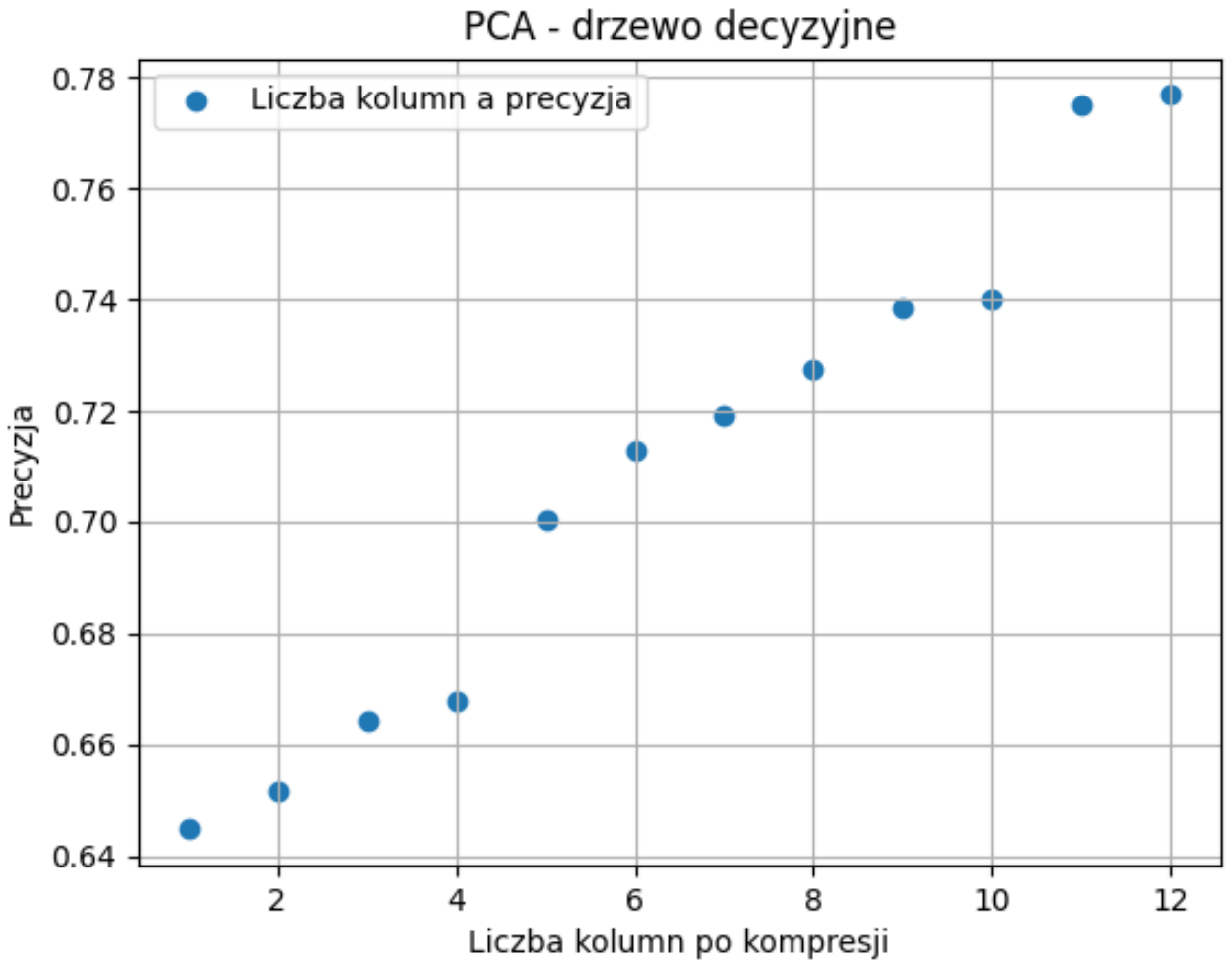
366	63
105	66

Precyzja: 72%

Brak wpływu standaryzacji

Średnia precyzja po 100 próbach dla 5 kolumn: 70.68%





Drzewo decyzyjne - PCA

Do badania PCA obliczyłem średnią z 50 próbek dla każdej kompresji.

KNN

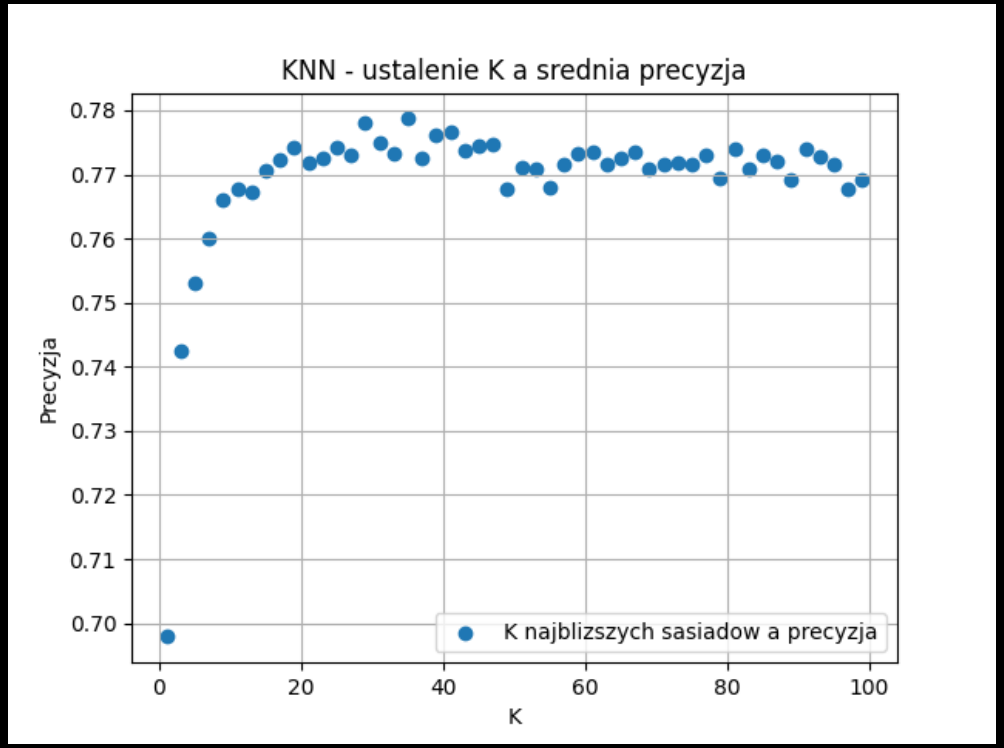
Aby określić jakie k jest najlepsze w tym klasyfikatorze dla moich danych wykonałem 50 próbek dla każdego nieparzystego k od 1 do 99.

Najlepszą precyzję osiągnął k = 35.

Przykładowa macierz błędów dla powyższego k:

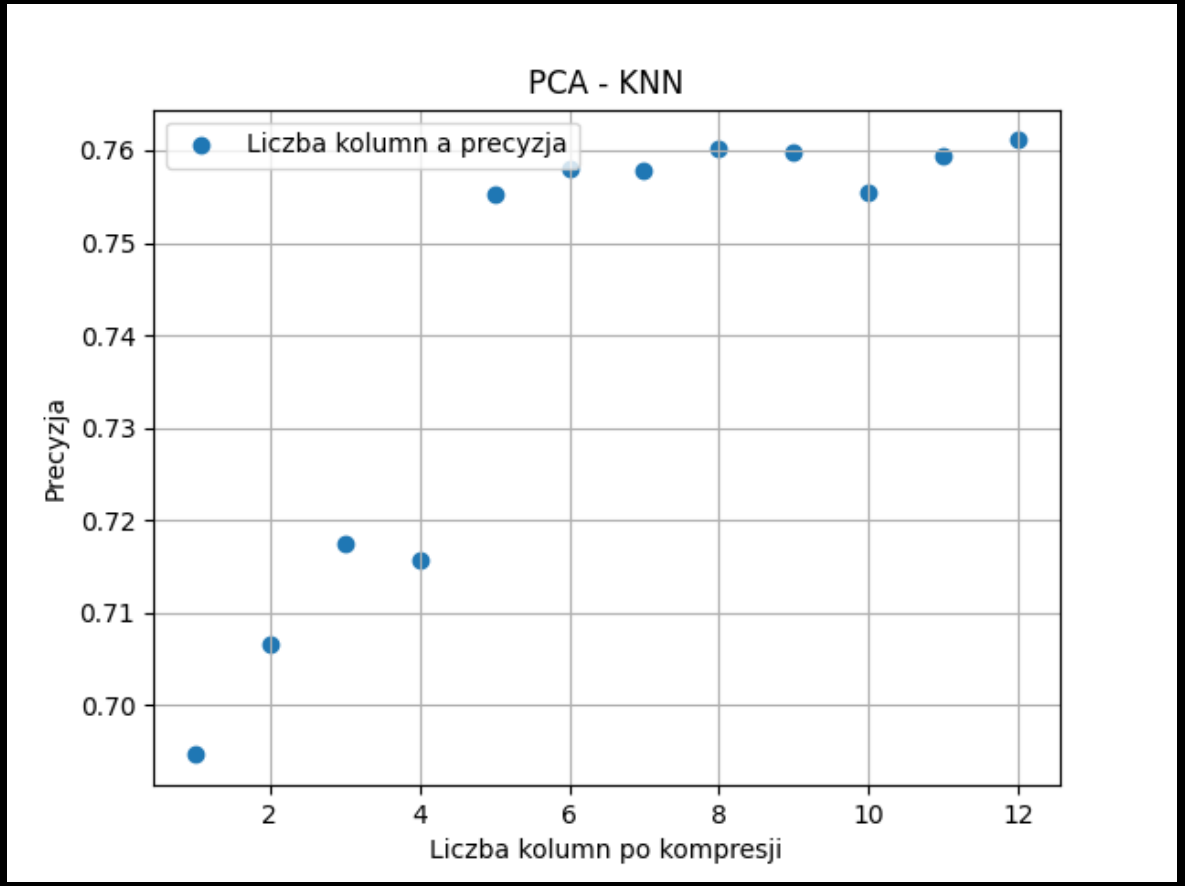
427	9
119	45

Precyzja: 78.67%



KNN – PCA i Standaryzacja

Wykres PCA: 50 próbek, k = 5



Do badania standaryzacji użyłem 100 próbek i k = 35 : Bez standaryzacji: 77.62%
Po standaryzacji: 78.35%

Naiwny Bayes

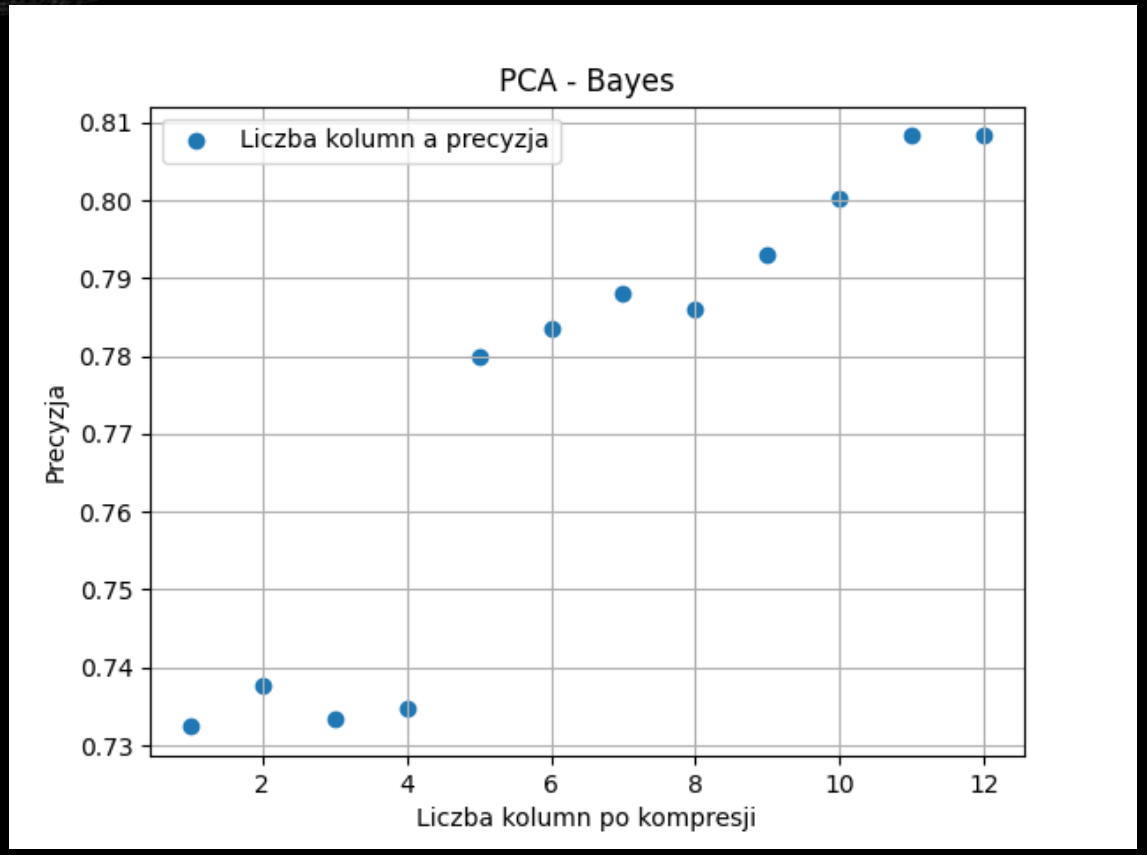
Średnia ze 100 próbek: 77.99%

Brak wpływu standaryzacji

Przykładowa macierz błędów:

419	19
111	51

Precyzja: 78.33%

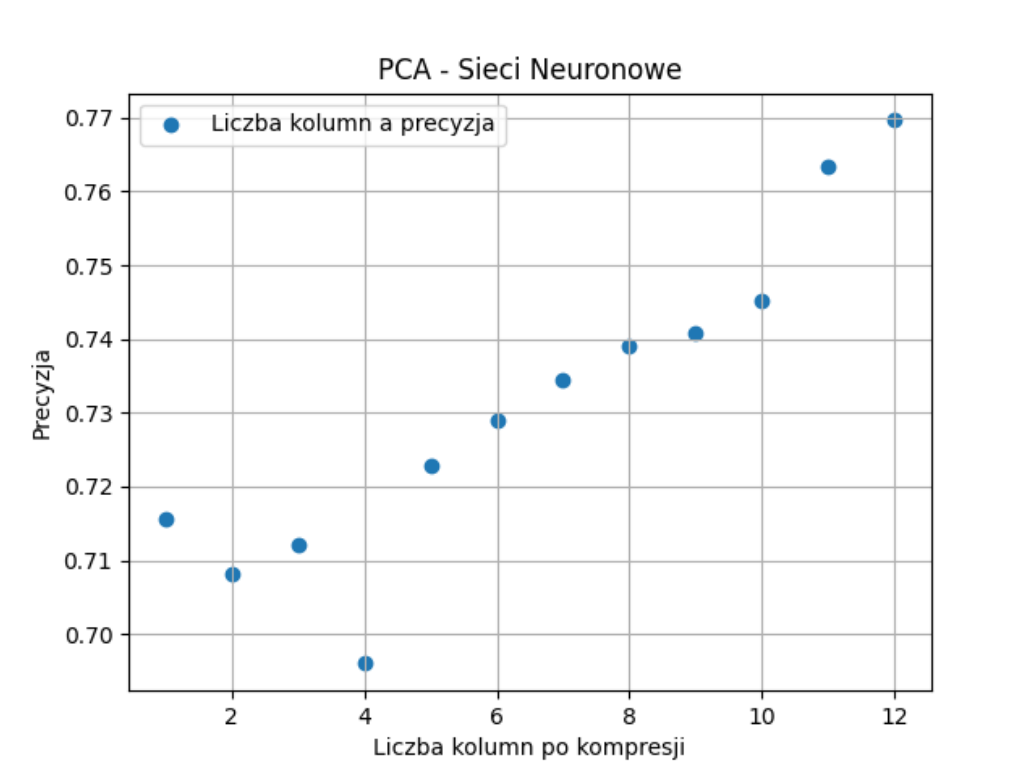


Sieci neuronowe

BADANIA PRZEPROWADZAŁEM NA SIECIACH O DWÓCH UKRYTYCH WARSTWACH, PO 64 UKRYTE NEURONY KAŻDA. **MAKSYMALNA LICZBA ITERACJI TO 10 TYSIĘCY - 1 MILION.**

DLA KAŻDEGO TESTU WYKONANE ZOSTAŁO 10 PRÓBEK O TAKICH SAMYCH PARAMETRACH.

Sieci neuronowe – PCA i Standaryzacja



Dla badań standaryzacji wyszły następujące wyniki:

Bez standaryzacji: 71.43%

Po standaryzacji: 74.01%

Sieci neuronowe – batch size i learning rate

Najlepszy batch size według badań: 16

Najlepszy learning rate według badań: 0.0425

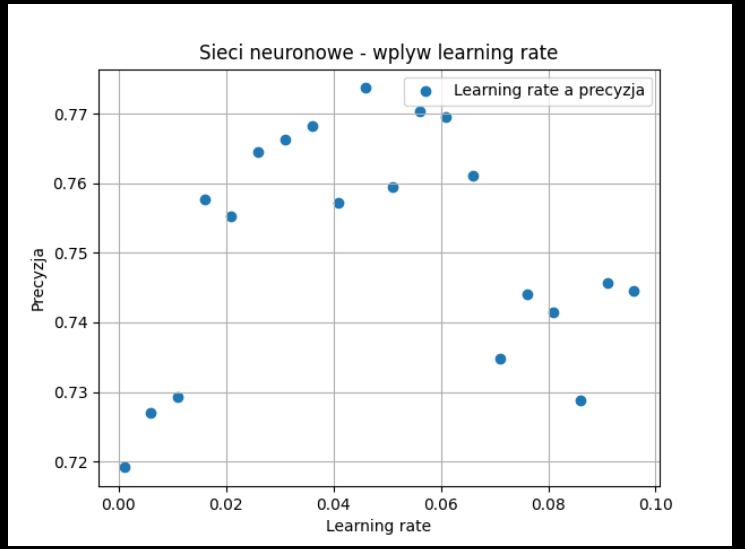
Korzystając z najlepszych zbadanych parametrów przeprowadziłem 10 próbek na tak skonfigurowanej sieci neuronowej.

Średnia precyzja: 78%

Najlepsza precyzja: 81%

Macierz błędów dla tej sieci:

429	9
105	57



Podsumowanie

Biorąc pod uwagę przeprowadzone badania, najlepszym klasyfikatorem jeśli chodzi o:

- Pojedynczy wynik precyzji była sieć neuronowa (5 kolumn, po standaryzacji, 10_000 iteracji, batch_size = 16, learning_rate = 0.0425)– 81%
- Najwyższa średnia z dużej liczby próbek - KNN dla k = 35 po standaryzacji: 78.35%
- Najlepsza dystrybucja FP i FN (najmniej fałszywie określonych gier jako wygrane względem fałszywie określonych gier jako przegrane) - drzewo decyzyjne: 105FP względem 63FN

Dziękuję za uwagę

Marcin Wojciechowski