

# Metody Rozpoznawania Obrazów I Podstawy Uczenia Maszynowego

## *Condensed Nearest Neighbours*

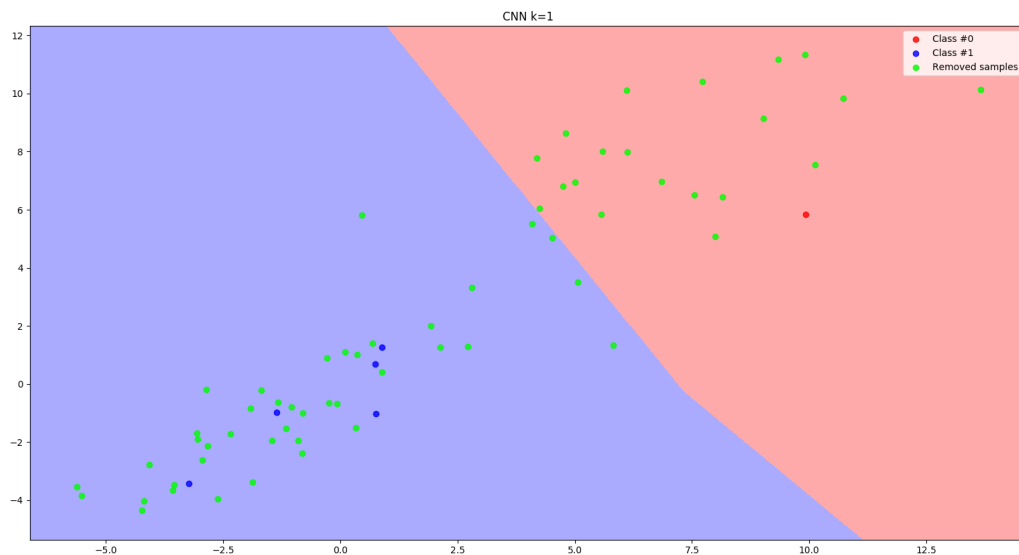
Autor: Ryszard Sikora

### 1 O danych testowych

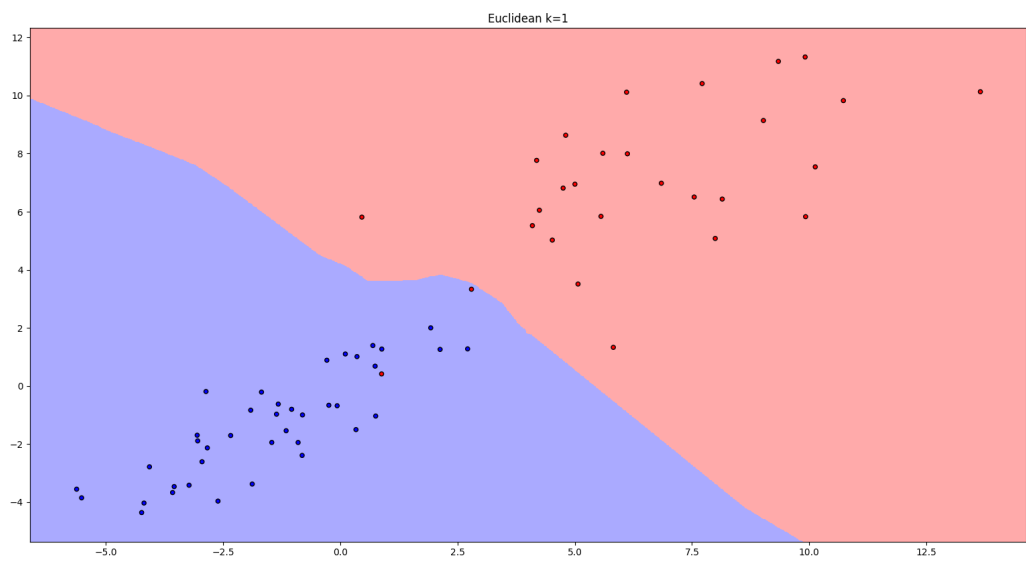
Wygenerowane dane dla obu prób składały się z dwóch klas po 50 egzemplarzy każda. Dane pochodziły z rozkładu normalnego o zadanej średniej i macierzy kowariancji.

### 2 Próba pierwsza

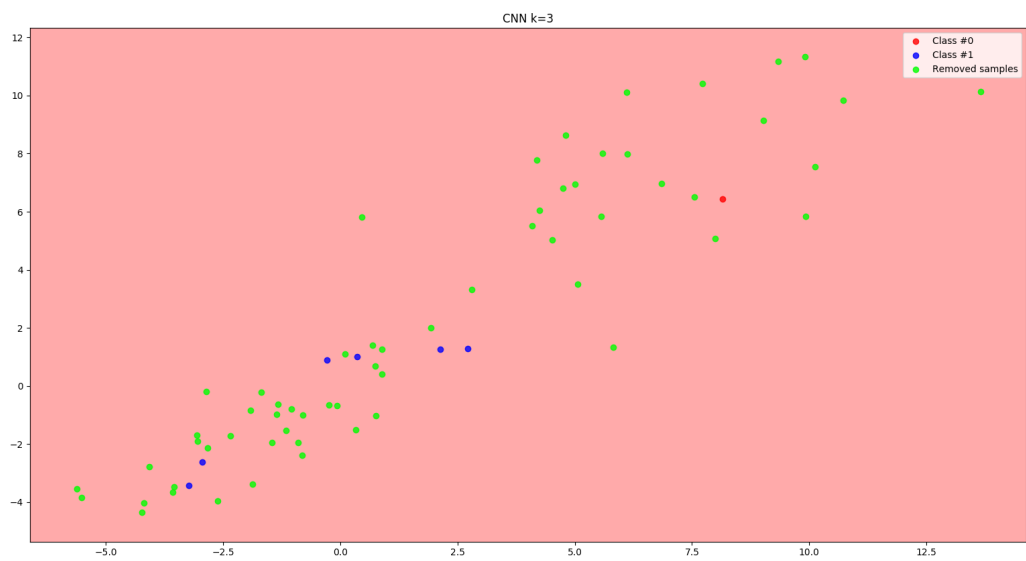
CNN,  $k=1$



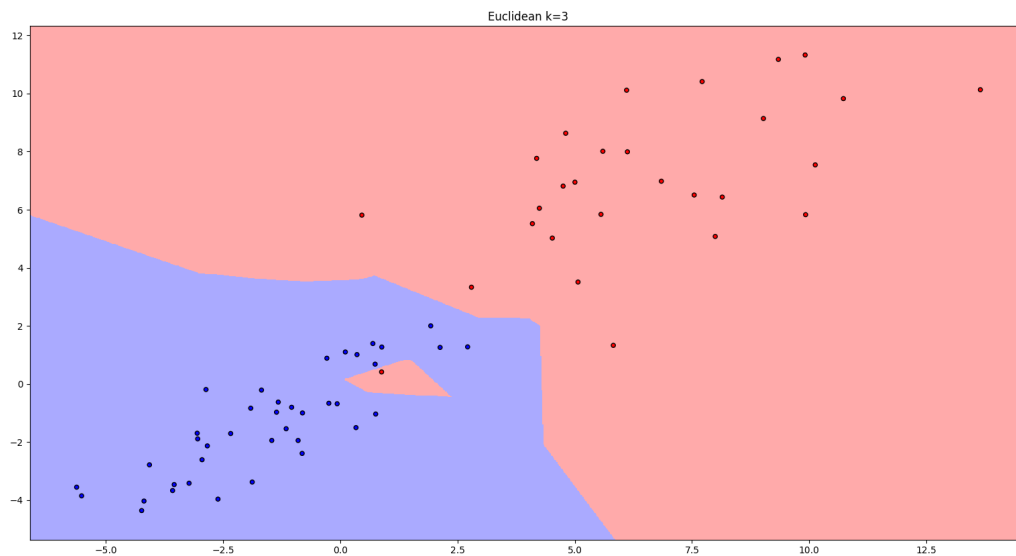
Euclidean, k=1



CNN, k=3



## Euclidean, k=3



Pierwszym co rzuca się w oczy jest to, że CNN dla k=3 to jakaś porażka okrutna. Być może algorytm trzeba było sparametryzować (jak było napisane w treści zadania), ale nie udało mi się znaleźć żadnej parametryzacji poprawiającej ten beznadziejny wynik. Dokumentacja biblioteki również milczy na ten temat.

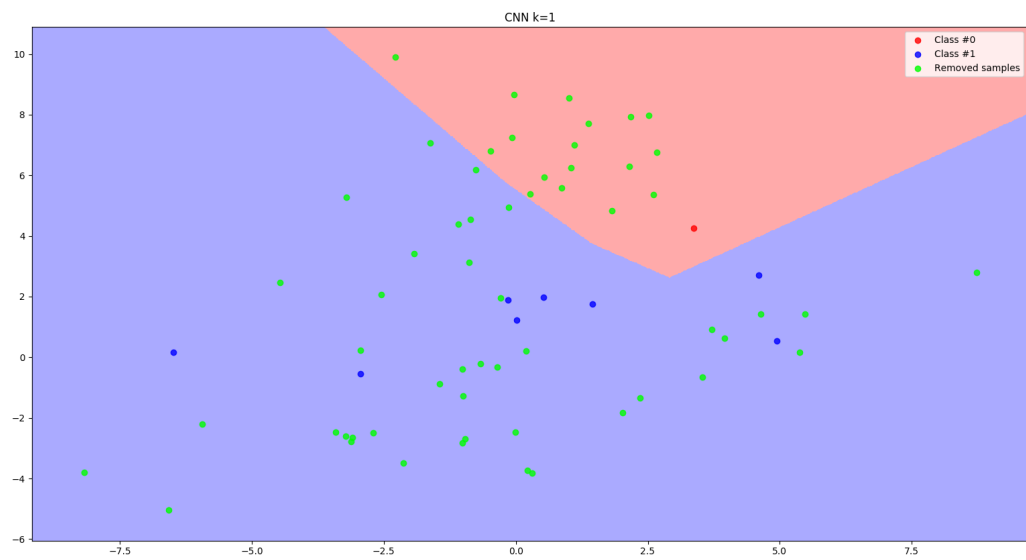
Wygląda na to, że dla k=1 CNN poradził sobie całkiem nieźle. Co prawda granica wygląda inaczej, ale, jak widać niżej, na testach różnica była niewielka:

CNN k=1	0.79
CNN reduction	0.089
Euclidean k=1	0.97
CNN k=3	0.34
CNN reduction	0.10
Euclidean k=3	0.94

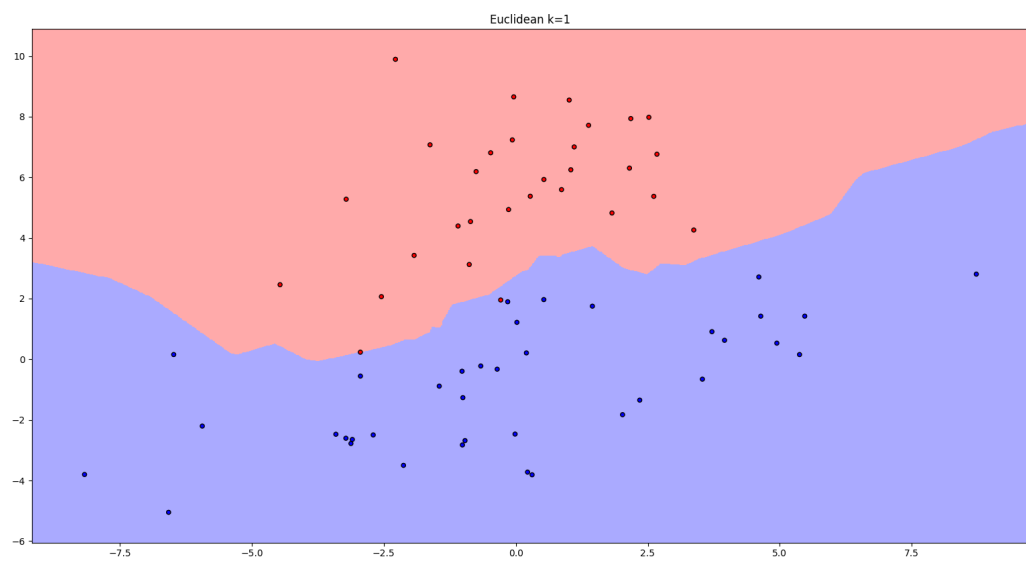
Redukcja określa jaką część danych uczących pozostała w zbiorze uczącym po kompresji.

## 3 Próba druga

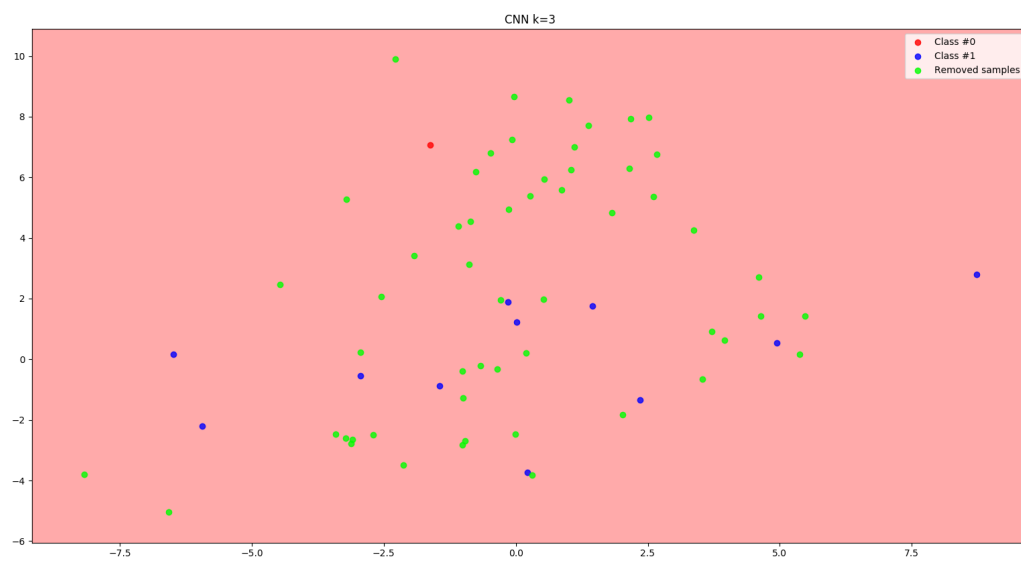
CNN, k=1



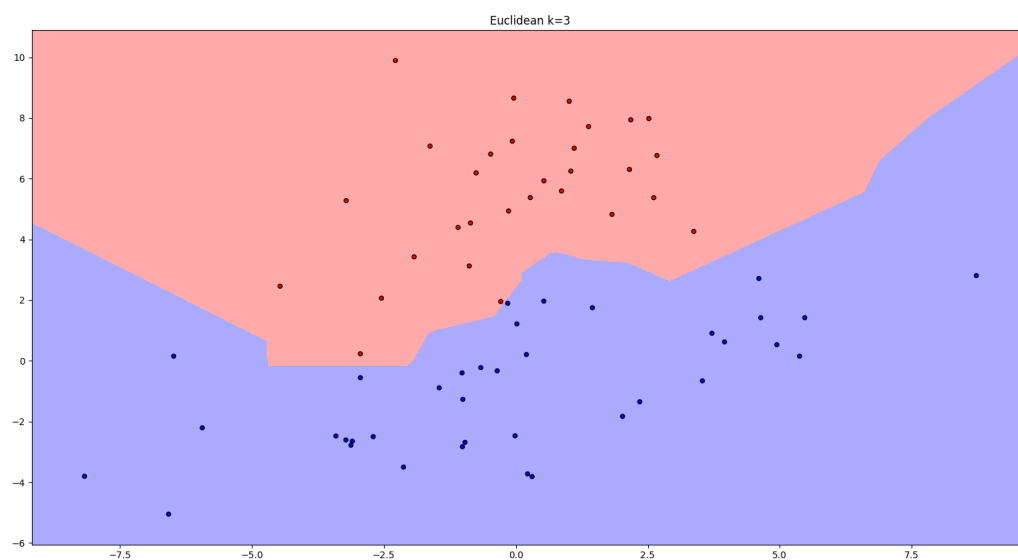
Euclidean, k=1



CNN,  $k=3$



Euclidean,  $k=3$



CNN z  $k=3$  dalej pozostaje bez komentarza. Natomiast dla tego zestawu danych pogorszył się również wynik CNN z  $k=1$ . Algorytm najwyraźniej może dawać bardzo różną dokładność w zależności od zestawu danych.

Poniżej jeszcze dokładność klasyfikacji:

CNN k=1	0.64
CNN reduction	0.13
Euclidean k=1	0.94
CNN k=3	0.40
CNN reduction	0.18
Euclidean k=3	0.97

Co ciekawe klasyfikator z metryką euklidesową i  $k=3$  miał większą dokładność od tego z  $k=1$  i tą samą metryką. Być może wiele punktów ze zbioru testowego leżało na granicy i akurat ułożyły się na korzyść większego wygładzenia tej granicy.