Metody rozpoznawania obrazów i podstawy uczenia maszynowego

Zadanie: PCA i kernel trick

Wykonanie: Kamil Kurp

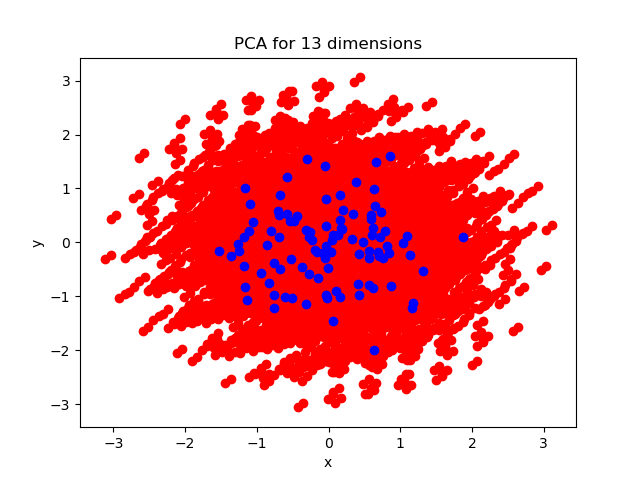
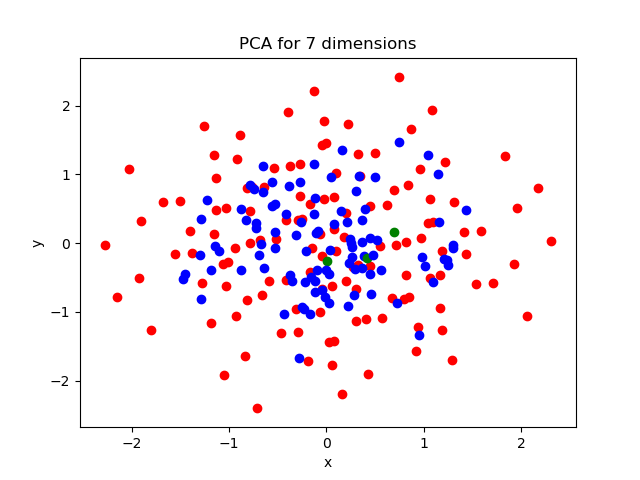
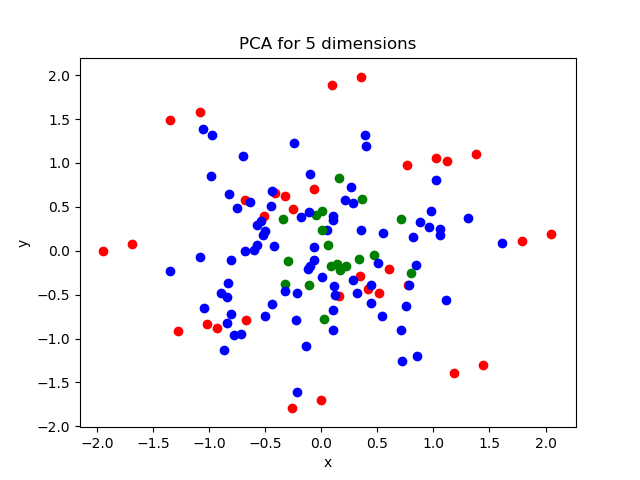
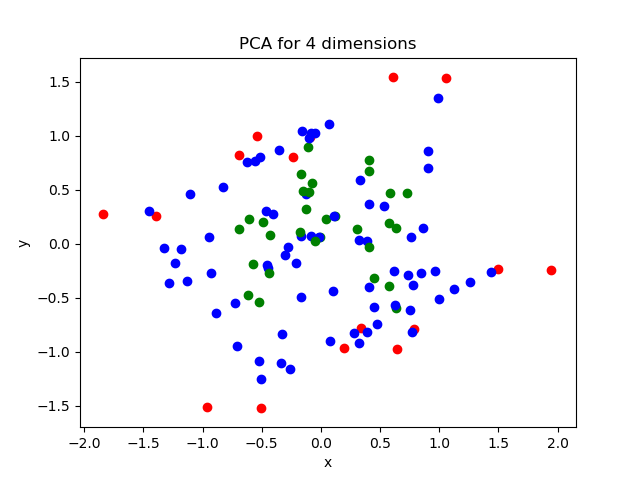
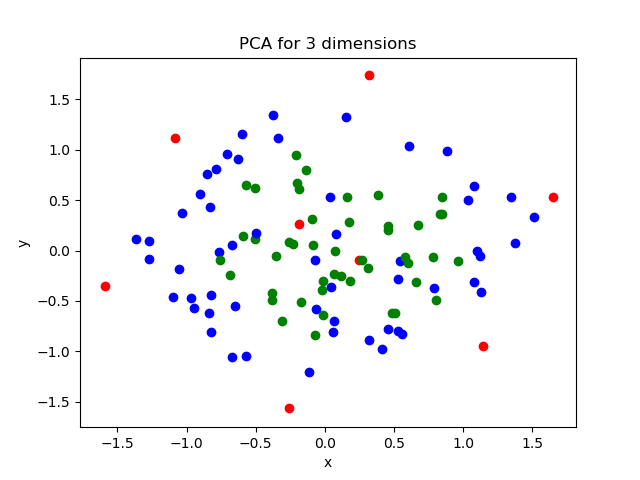
A:

Hiperkula wpisana w sześcian – rzut na 2D

Punkty wewnątrz sześcianu (ale nie kuli) – niebieski

Punkty w kuli – zielony

Narożniki sześcianu – czerwony

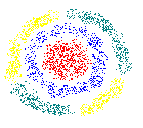


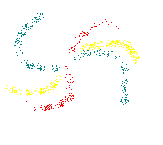
Wniosek:

PCA może posłużyć do rzutowania zbiorów obserwacji na przestrzenie o mniejszej ilości wymiarów. Na rysunkach łatwo zauważyć efekty klątwy wymiarów.

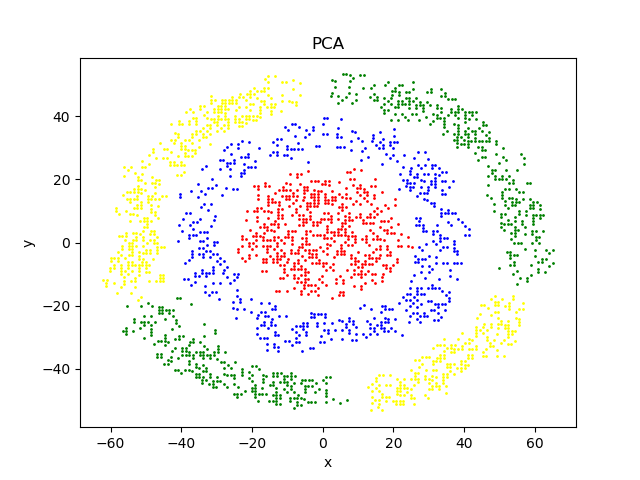
B:

Zbiory wejściowe:

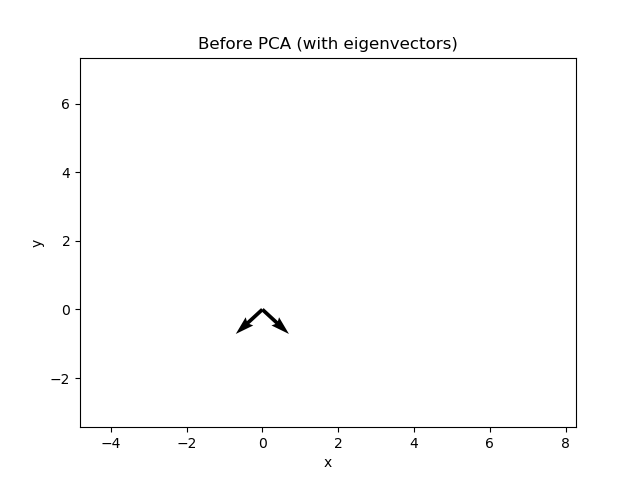




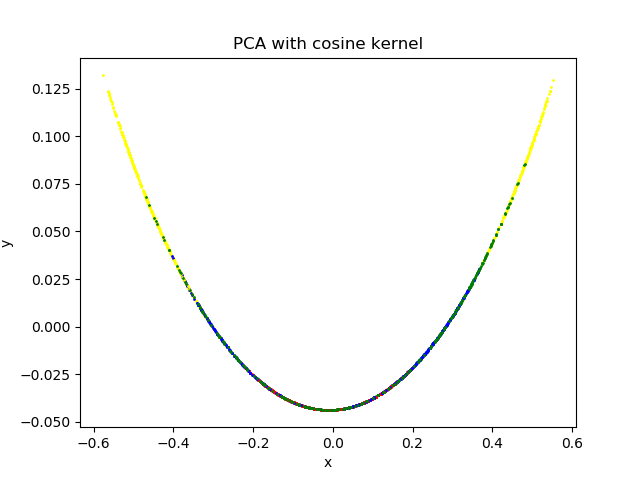
Zwykłe PCA (zbiór 1):



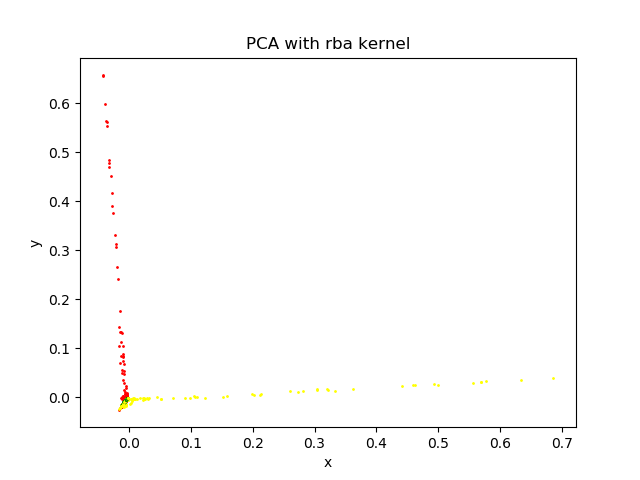
Wygląd wektorów własnych (zbiór 1):

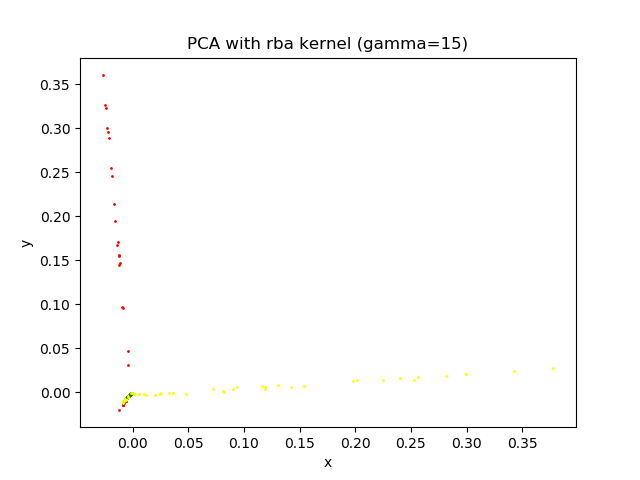
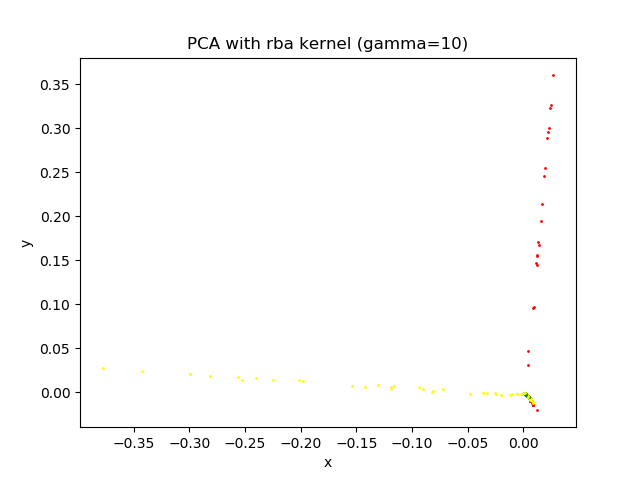
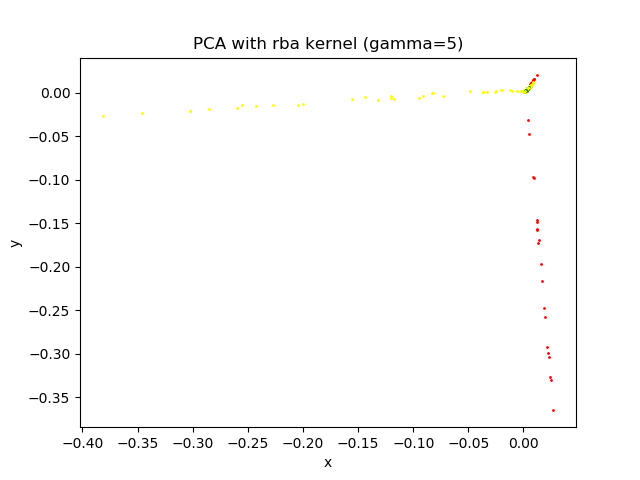


PCA z kerelem cosine (zbiór 1):

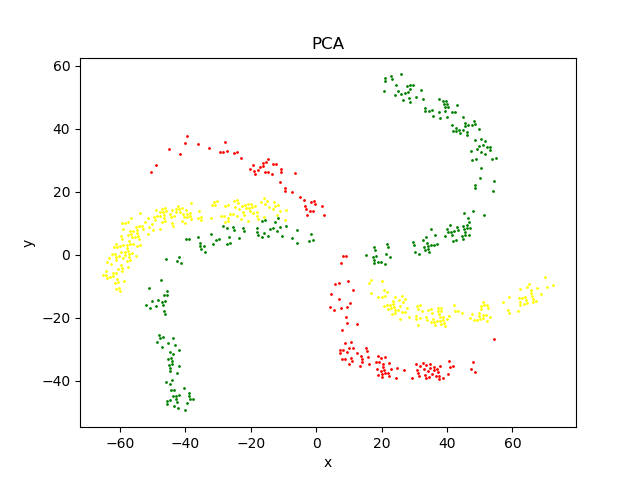


PCA z kernelem rbf (zbiór 1):

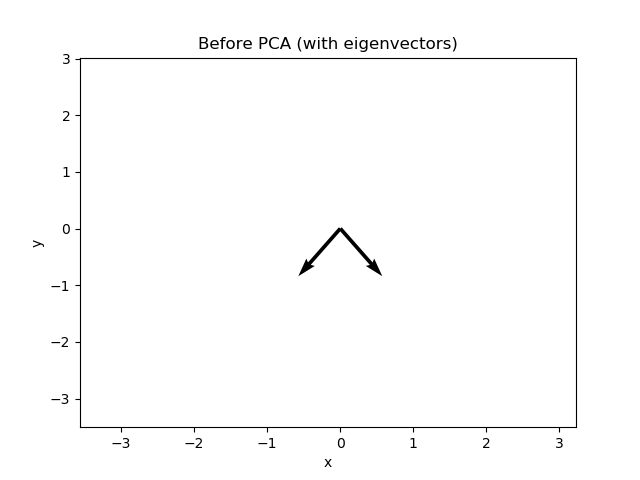




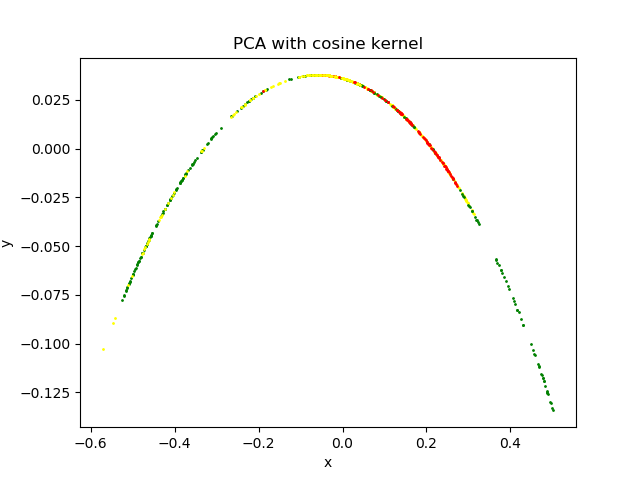
Zwykłe PCA (zbiór 2):



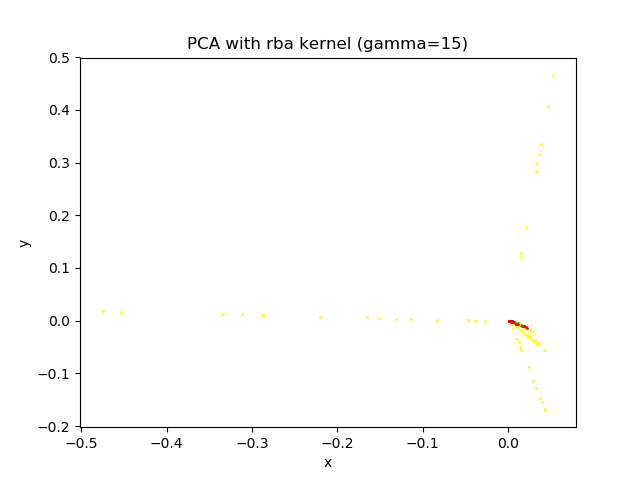
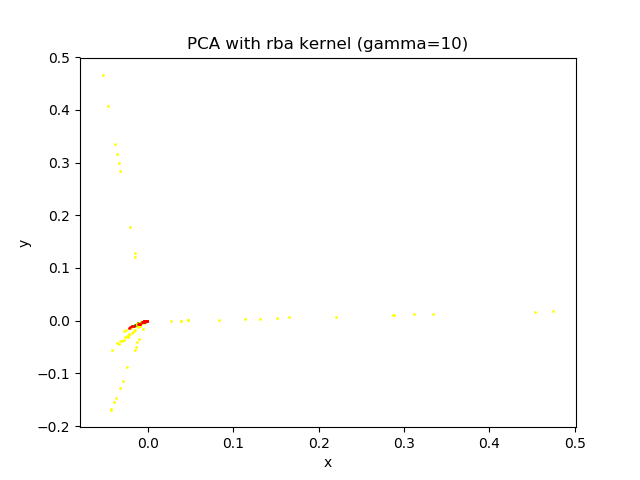
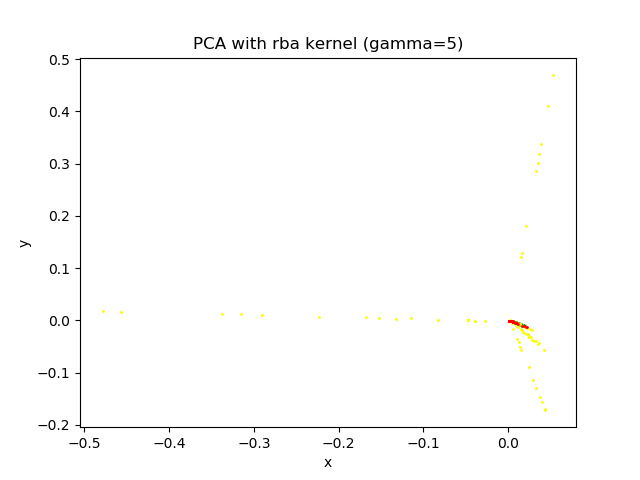
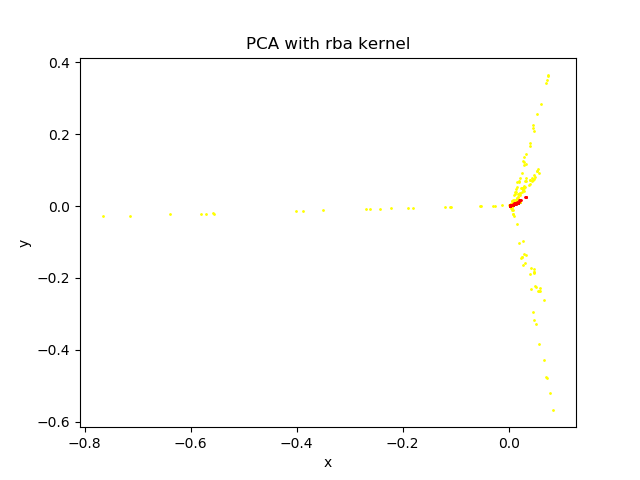
Wygląd wektorów własnych (zbiór 2):



PCA z kerelem cosine (zbiór 2):):



PCA z kernelem rbf (zbiór 2):



Wnioski:

Wykonanie zwykłego PCA bardzo nieznacznie wpływa na przestrzeń obserwacji. Zastosowanie kernel tricku z kernelem cosine powoduje umieszczenie punktów przestrzeni na paraboli, na której przejście między klasami punktów jest dosyć płynne. Zmiany współczynnika gamma powodują, że przestrzeń obraca się o 180 stopni i jest odwrócona (lustrzane odbicie).