

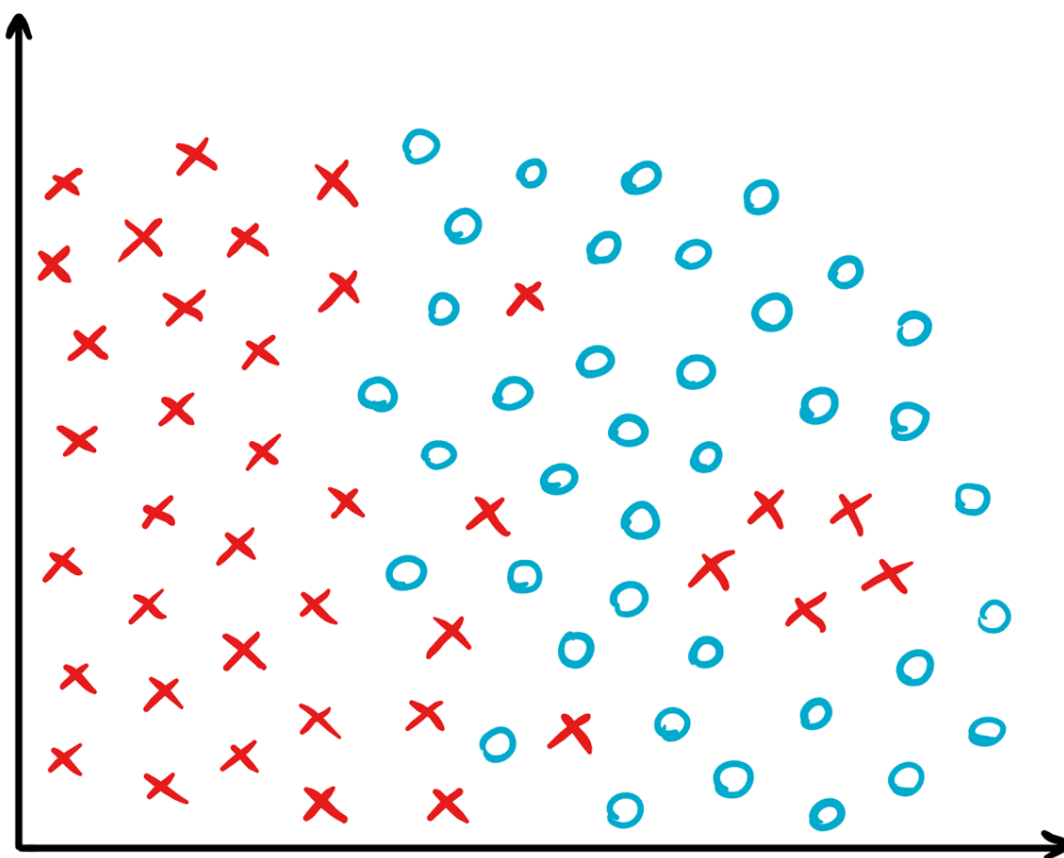
Zadanie 03

Tak na marginesie

Celem zadania będzie poeksperymentowanie z różnymi ustawieniami klasyfikatora SVM.

Dane do wykorzystania

Do wykonania zadania potrzebny jest niewielki, dwuwymiarowy zbiór danych. Zależy nam, by rozkład elementów był podobny do tego na poniższym obrazie - czyli nieregularny, z wyspami jednej klasy wewnątrz drugiej.



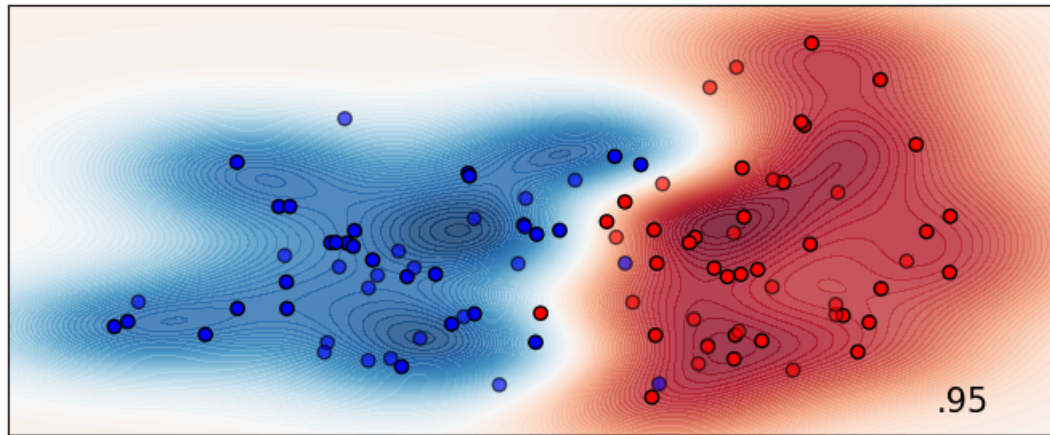
Jeżeli nie wiemy jak się za to zabrać, to możemy po prostu oznaczyć kolorami pojedyncze piksele w dowolnym edytorze grafiki rastrowej, a następnie przetworzyć taki obraz, odczytując położenie nie-białych punktów.

To co właściwie jest do zrobienia?

Kilka spraw.

- Po pierwsze - potrzebujemy być w stanie wizualizować kilka kwestii.
 - Jak wygląda granica decyzyjna - które obszary klasyfikator przypisuje do klasy czerwonej, a które do niebieskiej (i z jakim poziomem pewności). Może się tu mocno przydać kod ze sztanowego przykładu z biblioteki *scikit-learn*: https://scikit-learn.org/stable/auto_examples/classification/plot_classifier_comparison.html - warto jednak zwiększyć rozdzielczość skali kolorów (by wykrywanych było więcej poziomów). Wynik może wyglądać mniej-więcej tak jak poniżej (kreatywność i odbieganie od domyślnej stylistyki

oczywiście mile widziane).



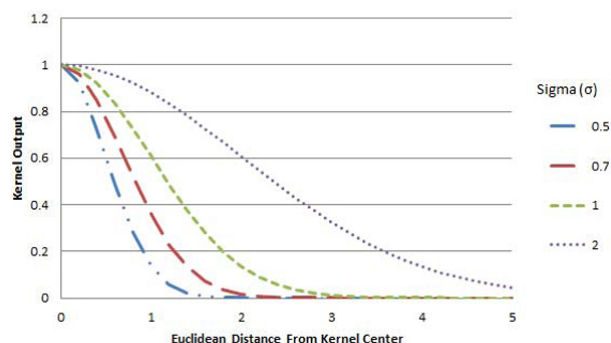
- Które obserwacje ze zbioru wejściowego zostały wybrane jako wektory wspierające.
- Jaka jest aktualna szerokość marginesu między klasami (i jej zależność od danego parametru) - przynajmniej dla liniowego SVM.
- Po drugie - chcemy użyć powyższych narzędzi do sprawdzenia, jak na naszym zbiorze spiszą się poniższe warianty SVM.
 - "Zwykły" liniowy SVM...
 - ...z dużym naciskiem na próbki po "właściwej" stronie (wysokie "kary" za nieprawidłowości)...
 - ...oraz z dużym naciskiem na regularyzację i szeroki margines (niskie "kary").
 - Warto tu sprawdzić więcej niż dwie wartości parametrów, tylko ich szersze spektrum - w tym opcje "skrajne".
 - Kernel SVM - czyli taki z iloczynem skalarnym podmienionym na...
 - ...iloczyn pierwszej składowej obu wektorów, pomijający drugą składową;
 - ...klasyczny kernel typu RBF (*Radial Basis Function*);
 - Opisuje go wzór taki jak na obrazku poniżej.

$$K(\mathbf{x}, \mathbf{x}') = \exp\left(-\frac{\|\mathbf{x} - \mathbf{x}'\|^2}{2\sigma^2}\right)$$

- Jest to funkcja zależna od tytułowego promienia - normy z odległości między wejściowymi wektorami.
- Na jej pracę wpływa też czynnik normalizujący *sigma*. Warto sprawdzić skutki użycia omawianego kernela dla szerokiego spektrum wartości tego parametru.

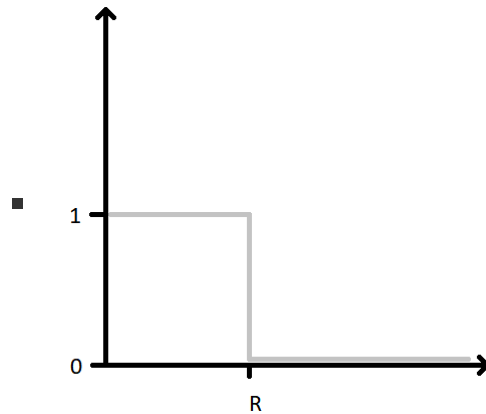
■

Gaussian Radial Basis Function Kernel

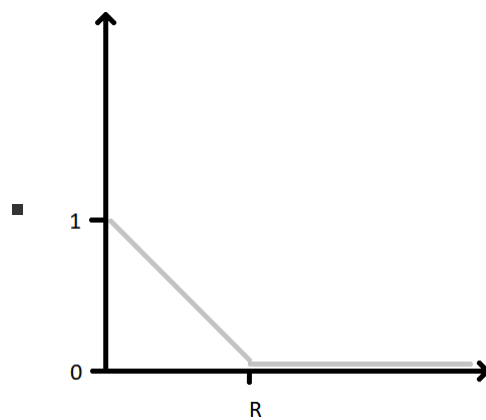


- ..."kanciasty" kernel typu RBF: funkcję, która zwraca 1 jeżeli wektory są odległe o mniej niż pewna ustalona minimalna odległość R ("leżą koło siebie") i 0 w przeciwnym wypadku ("nie leżą w pobliżu");

- Podobnie jak w przypadku wspomnianego w poprzednim punkcie parametru σ - warto zbadać zachowanie dla różnych wartości R , dobierając rozważany ich zakres w zależności od typowych odległości w zbiorze danych.



- ..."częściwo liniowy" kernel typu RBF: funkcję, która poniżej pewnej minimalnej odległości R zwraca $(R - \text{odległość})/R$ - a dla mniejszych zwraca po prostu 0;



- ...kernel zależny od kąta między wektorami prowadzącymi od punktu leżącego w centrum zbioru danych do poszczególnych obserwacji ("wskazówek zegara").
 - Powinien się zachowywać jak klasyczne podobieństwo cosinusowe - zwracać 1 dla równoległych wektorów, 0 dla prostopadłych.
- Dla każdego z tych wariantów obserwujemy jak "pokolorowana" jest przestrzeń decyzyjna - jak zmiana parametrów (lub podmiana iloczynu skalarnego) wpłynęła na rezultaty.