

# TWM C3:Uczenie maszynowe, klasyfikator SVM

Jacek Orliński, Jakub Kulawik

## Zadanie:

1. Uruchom klasyfikator SVM używający jądra Gaussa na tych danych w celu klasyfikacji sceny na obrazie. Klasyfikator powinien przyjmować na wejściu obraz i wybierać jedną z trzech dostępnych klas.
2. Zaimplementuj samodzielnie optymalizację parametrów C i gamma klasyfikatora SVM na stworzonych zbiorach: uczącym, walidacyjnym i testowym
  1. Zastosuj metodę wielokrotnej kross-walidacji (k-fold cross-validation) w celu określenia hiperparametrów na zbiorze uczącym i walidacyjnym, określ odpowiednie k
  2. Dobierz parametry C i gamma klasyfikatora stosując wyczerpujące przeszukiwanie grid-search w celu określenia zestawu parametrów dającego największą skuteczność na zbiorach walidacyjnych (patrz np. <https://towardsdatascience.com/hyperparameter-tuning-for-support-vector-machines-c-and-gamma-parameters-6a5097416167>)
  3. Zareportuj szczegółowo wyniki przeszukiwania (np. za pomocą tabeli), zwizualizuj przestrzeń parametrów i odpowiadającą im skuteczność rozpoznania w 3D
3. Wykorzystaj gotowe narzędzia do określenia optymalnego zestawu parametrów - porównaj z otrzymanymi samodzielnie wynikami.

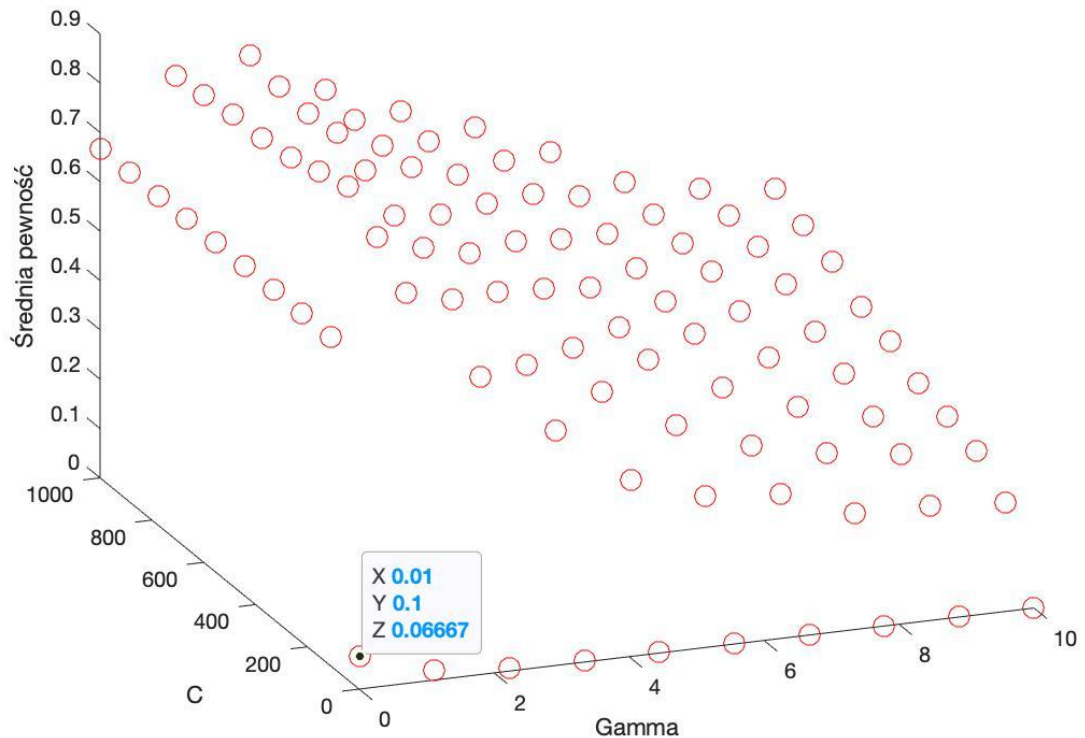
W zadaniu zdecydowano się wybrać 'deli' jako wykrywany zestaw zdjęć. Wykrywanie zostało oparte o histogramy wygenerowane przez udostępniony kod. Ponieważ SVM rozwiązuje problemy binarne, do rozróżnienia trzech kategorii obrazów wykorzystano dwie maszyny (jedna sprawdzająca *deli* vs *greenhouse*, druga *deli* vs *bathroom*). Wyniki były następnie syntetyzowane za pomocą porównania odgadnięć obu maszyn:

- *Deli* jeżeli obie maszyny na to wskazują
- Jeżeli żadna maszyna nie wskazała *deli*, wynik jest arbitralny między *greenhouse* a *bathroom*. Błąd ten jest popełniany świadomie; w wypadku gdyby celem zadania było rozróżnienie wszystkich trzech kategorii, można by go zniwelować poprzez implementację trzeciego SVM
- W przeciwnym wypadku (*bathroom* lub *greenhouse* vs *deli*) wybierana jest kategoria niebędąca *deli*

Utworzono grid-search rozpatrujący parametry Gamma i C w zakresie Gamma(0,10) i C(0,1000). Obie maszyny są obsługiwane oddzielnie, ale podczas jednego przeszukania.

Przeszukiwanie parametrów obu maszyn jest podobne, więc zamieszczono tylko po jednej ilustracji.

## 1. Dopasowanie parametrów poprzez grid-search



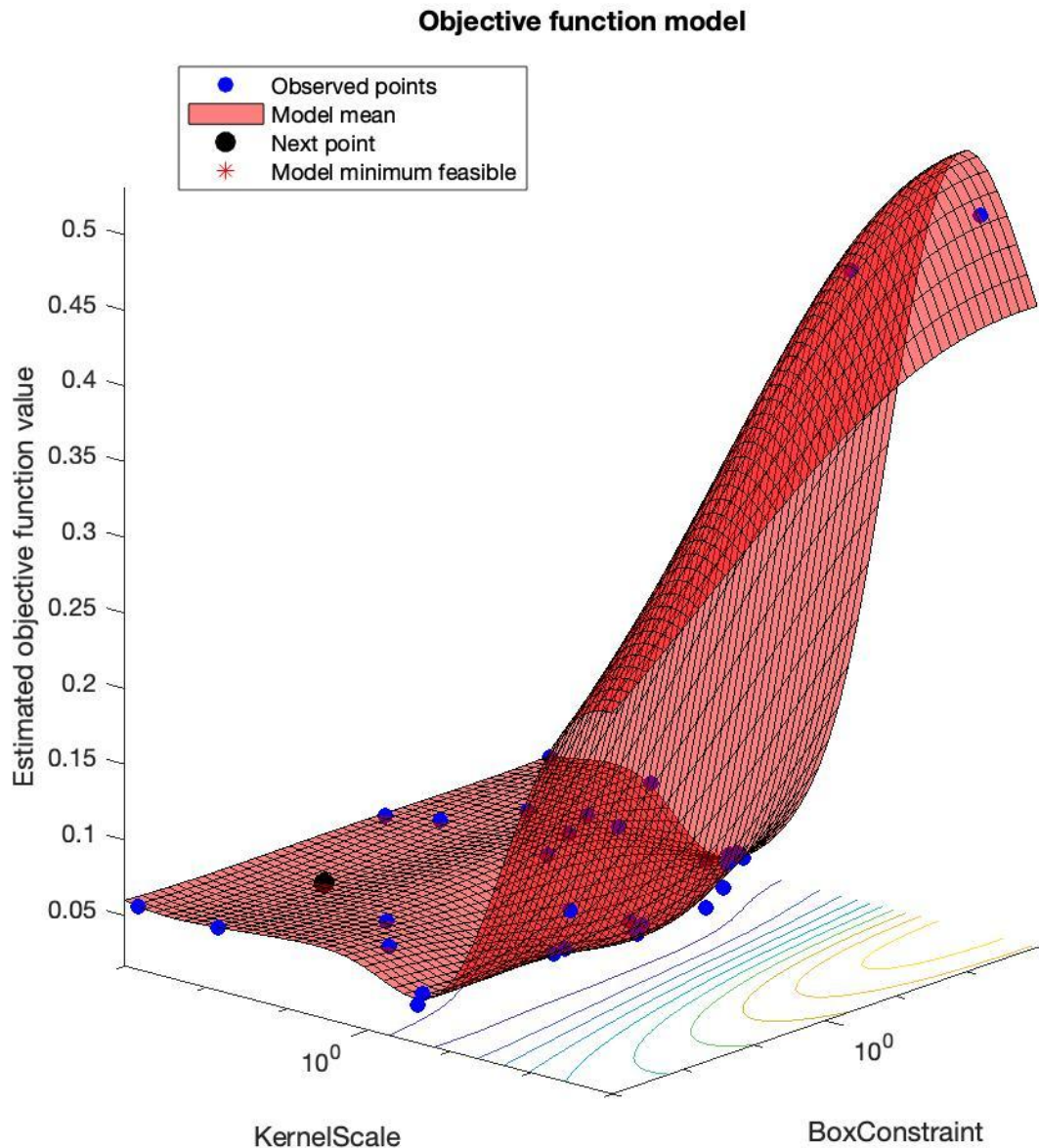
Parametry dopasowano poprzez wyczerpujące przeszukiwanie kratownicy możliwych kombinacji w zadanym zakresie. W celu przeszukania kratownicy tworzone nowe modele SVM, których wyniki były oceniane w funkcji celu. Zbiór treningowy miał 60 obrazów każdej kategorii.

Zdefiniowano funkcje celu jako średnią ocen negatywnych zwróconych przez klasyfikator. Średnia została przemnożona przez maskę będącą tablicą składającą się z -1 i 1, która zmieniła znak dla obrazów które miały być zakwalifikowane pozytywnie. Dzięki temu obrazy które miały być zaklasyfikowane jako pozytywne, a zostały jako negatywne zwracały punkty ujemne, i te które zostały poprawnie zaklasyfikowane jako negatywne zwracały minusowe wartości. W związku z tym dobór parametrów polegał na minimalizowaniu funkcji celu.

Przeszukano kilka kratownic o zadanych zakresach parametrów. Wśród nich było kilka kratownic o większej liczbie punktów oraz ich logarytmicznym rozkładzie; pomimo znalezienia różnych rozwiązań, dających lepsze wartości funkcji celu, rozpoznawanie obrazów pozostawało zbliżonym poziomem i różniło się np. przemieszczeniem jednego błędu w inne miejsce macierzy pomyłek.

## 2. Dopasowanie automatyczne

W celu dopasowania poprzez automat ustawiono w matlabie automatyczne dobieranie hiperparametrów.



Automatyczne przeszukiwanie sprawdziło szerszy zakres parametrów niż nasza implementacja grid-search, więc znalazła również inne rozwiązania. Odgadnięcia obu skalibrowanych automatycznie maszyn SVM były łączone podobnie jak w metodzie grid search.

### 3. Porównanie obu metod

Rozwiązanie sprawdzono testując 15 obrazów każdej kategorii.

**SVM z doбором parametrów poprzez grid search**

True Class	bathroom	deli	greenhouse
	10	4	1
		15	
Predicted Class	bathroom	deli	greenhouse
		2	13

**SVM z doбором automatycznym**

True Class	bathroom	deli	greenhouse
	13	1	1
		14	
Predicted Class	bathroom	deli	greenhouse
		2	13

	<b>Grid search</b>	<b>Auto</b>
<b>Deli false negative</b>	0/15	1/15
<b>Deli false positive</b>	6	3
<b>Total deli errors</b>	6	4
<b>Overall error rate</b>	7/45	5/45

Jak widać w powyższej tabeli, rozwiązanie automatyczne okazało się odrobinę lepsze.

Uzyskane wyniki nie wynikają ze zbyt restrykcyjnego zakresu parametrów; wyniki zatrzymywały się w tych samych miejscach mimo zwiększenia zakresu do odpowiadającemu wynikom dopasowania automatycznego. Zwiększanie liczby punktów w kratownicy również tego nie zmieniło. Sposób przeszukiwania (zamiana kolejności zagnieżdżonych pętli) również nie miała wpływu na wynik.

Zauważalną zmianę w znalezionych parametrach przyniosła jedynie zmiana kratownicy na rozmieszczoną logarymicznie; jednakże pomimo innych parametrów, odgadnięcia systemu pozostawały niemalże identyczne.

Ze względu na powyższe można więc przypuszczać, że grid search znajduje inne minimum lokalne niż metoda automatyczna, a że nie szuka innych minimów, blokuje się w tym samym rozwiązaniu. Możliwe jest również, że zdefiniowana funkcja celu jest gorsza od automatycznej.