Maszyna wektorów nośnych

Kamil Łangowski
Wydział Fizyki Technicznej i Matematyki Stosowanej
Politechnika Gdańska

15 czerwca 2021

1 Teoria

Maszyna wektorów nośnych (ang. *support vector machine*, SVM) to model nadzorowanego uczenia maszynowego, który znajduje zastosowanie w zadaniach związanych zarówno z klasyfikacją, jak i regresją. Omawiając SVM w głównej mierze skupimy się na zastosowaniach w problemach klasyfikacji. Zanim jednak zajmiemy się maszyną wektorów nośnych, w celu zapobiegnięcia niedomówień, musimy wprowadzić model zwany klasyfikatorem maksymalnego marginesu (ang. *maximal margin classifier*). Następnie uogólnimy go otrzymując klasyfikator wektorów nośnych (ang. support *vector classifier*), który z kolei uogólnia się do maszyny wektorów nośnych. Często można spotkać się z sytuacją, w której to każdy z przytoczonych powyżej modeli występuje pod nazwą "maszyna wektorów nośnych".

1.1 Klasyfikator maksymalnego marginesu

Konstruując klasyfikator maksymalnego marginesu bazować będziemy na pojęciu hiperpłaszczyzny, tzn. $d-1$ wymiarowej podprzestrzeni afinicznej pewnej przestrzeni d-wymiarowej (np. hiperpłaszczyzna w przestrzeni 2-wymiarowej jest krzywą).

Definicja 1 (Hiperpłaszczyzna) *Hiperpłaszczyzna w przestrzeni d-wymiarowej dla obserwacji* $x = (x^{(1)}, x^{(2)}, \ldots, x^{(d)})$ *opisana jest przez równanie*

$$\beta_0 + \sum_{i=1}^{d} \beta_i x^{(i)} = 0, \tag{1}$$

gdzie $\beta_0, \beta_1, \ldots, \beta_d$ *są parametrami równania.*

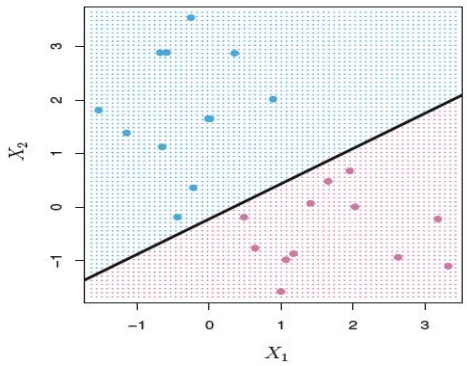
Jeżeli pewna obserwacja $x = (x^{(1)}, x^{(2)}, \ldots, x^{(d)})$ spełnia równanie (1), wówczas punkt ten znajduje się na rozważanej hiperpłaszczyźnie. Natomiast, jeżeli x nie spełnia (1) oraz zachodzi

$$\beta_0 + \sum_{i=1}^{d} \beta_i x^{(i)} < 0, \tag{2}$$

oznacza to, że x znajduje się po jednej ze stron hiperpłaszczyzny. W analogiczny sposób określone jest położenie punktu x, po drugiej ze stron. Mianowicie

$$\beta_0 + \sum_{i=1}^{d} \beta_i x^{(i)} > 0. \tag{3}$$

Widzimy zatem, że hiperpłaszczyzna rozdziela d-wymiarową przestrzeń na dwie odrębne części. Ponadto posiadając informacje na temat parametrów hiperpłaszczyzny, bez trudu możemy zdeterminować, po której ze stron znajduje się rozważany punkt.



Rysunek 1: Hiperpłaszczyzna rozdzielająca dwie klasy obserwacji. Na powyższym rysunku $X_1 = = X^{(1)}$ oraz $X_2 = X^{(2)}$.

Załóżmy, że X jest $n \times d$ wymiarową macierzą zawierającą n-wymiarowe obserwacje

$$x_1 = \begin{pmatrix} x_1^{(1)} \\ x_1^{(2)} \\ \vdots \\ x_1^{(d)} \end{pmatrix}^T, \ldots, x_n = \begin{pmatrix} x_n^{(1)} \\ x_n^{(2)} \\ \vdots \\ x_n^{(d)} \end{pmatrix}^T. \tag{4}$$

Ponadto niech każda z rozważanych obserwacji należy do jednej z dwóch klas $y \in \{-1, 1\} = Y$. Załóżmy także, że mamy pewną obserwację testową $x_j = (x_j^{(1)}, x_j^{(2)}, \ldots, x_j^{(d)})$. Podobnie jak w przypadku regresji logistycznej chcemy uzyskać klasyfikator, za pomocą którego będziemy mogli określić przynależność rozważanej obserwacji testowej. W tym celu posłużymy się pojęciem hiperpłaszczyzny separującej (ang. *separating hyperplane*).

Definicja 2 (Hiperpłaszczyzna separująca) *Hiperpłaszczyzna separująca to hiperpłaszczyzna, która jednoznacznie rozdziela obserwacje należące do różnych klas.*

Oznacza to, że obserwacje należące do klasy 1 znajdują się po jednej stronie hiperpłaszczyzny separującej, a obserwacje należące do klasy -1 po przeciwnej. Przykładowo

$$\begin{cases} \beta_0 + \sum_{i=1}^{d} \beta_i x_j^{(i)} > 0, & \text{gdy } y_j = 1 \\ \beta_0 + \sum_{i=1}^{d} \beta_i x_j^{(i)} < 0, & \text{gdy } y_j = -1 \end{cases} \tag{5}$$

Zatem, w przypadku istnienia hiperpłaszczyzny separującej, możemy z niej skorzystać w celu konstrukcji klasyfikatora binarnego. Niech $f(x_j) = \beta_0 + \sum_{i=1}^{d} \beta_i x_j^{(i)}$. w przypadku, gdy $f(x_j) > 0$ obserwacja x_j należy do klasy 1, natomiast dla $f(x_j) < 0$ obserwacja x_j należy do klasy -1. Zauważmy także, że jeśli $f(x_j)$ osiąga wartości bliskie zeru, to obserwacja x_j znajduje się bliżej hiperpłaszczyzny, analogicznie dla dalszych obserwacji. Przewidywaną klasę obserwacji testowej x_j możemy zapisać także jako

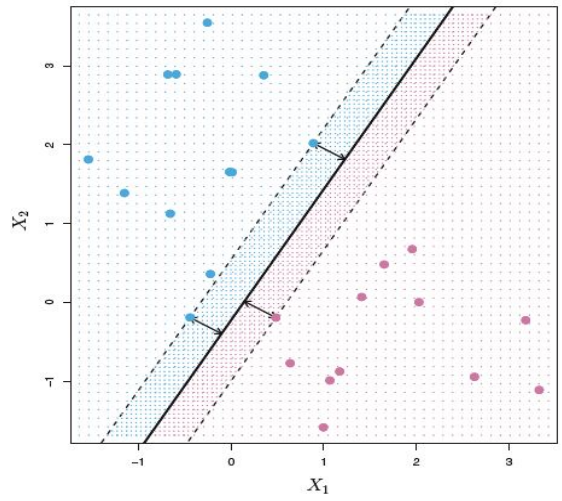
$$y_j = \text{sign}\left(f(x_j)\right), \tag{6}$$

gdzie sign jest funkcją znaku.

Jeśli dane jakim się zajmujemy mogą być doskonale rozdzielone poprzez hiperpłaszczyznę,

wtedy istnieje nieskończona liczba hiperpłaszczyzn separujących. Z tego względu należy wybrać hiperpłaszczyznę, która w najlepszy sposób będzie spełniała rolę klasyfikatora. Sposobem na wybór takiej hiperpłaszczyzny separującej jest hiperpłaszczyzna maksymalnego marginesu (ang. *maximal margin hyperplane*), która wyróżnia się tym, iż jest najbardziej oddalona od wszystkich obserwacji. Mając zadaną hiperpłaszczyznę rozdzielającą, wyznaczamy odległość punktów obserwacji od niej oraz wybieramy tę, która jest najmniejsza – nazywamy ją marginesem. Hiperpłaszczyzna maksymalnego marginesu to hiperpłaszczyzna, dla której margines jest największy. Za pomocą takiej hiperpłaszczyznę możemy klasyfikować obserwacje na podstawie ich położenia względem niej. Tak powstały klasyfikator nazywany jest klasyfikatorem maksymalnego marginesu. W takim podejściu do klasyfikacji zakładamy, że duży margines dla danych treningowych będzie także odpowiedni dla danych testowych.

W praktyce klasyfikator maksymalnego marginesu często działa poprawnie, jednak w przypadku obserwacji o dużym wymiarze ma skłonności do nadmiernego dopasowania. Na rysunku 2 przedstawiono przykładową hiperpłaszczyznę maksymalnego marginesu. Odległość od ciemnej linii – hiperpłaszczyzny separującej, do linii przerywanych jest marginesem hiperpłaszczyzny separującej. Zwróćmy uwagę, że trzy spośród najbliższych obserwacji do hiperpłaszczyzny maksymalnego marginesu są od niego równo oddalone. Te punkty nazywamy wektorami nośnymi (ang. *support vectors*), ze względu na fakt, iż przesunięcie jakiegokolwiek z nich skutkowałoby przesunięciem hiperpłaszczyzny separującej. Problem konstrukcji klasyfikatora maksymalnego marginesu sprowadza się do odpowiedniego wyboru parametrów równania hiperpłaszczyzny separującej w taki sposób, by margines utworzony za pomocą wektorów nośnych był największy.



Rysunek 2: Hiperpłaszczyzna maksymalnego marginesu. Na powyższym rysunku $X_1 = X^{(1)}$ oraz $X_2 = X^{(2)}$.

1.2 Klasyfikator wektorów nośnych

Jeżeli niemożliwym jest rozdzielenie obserwacji hiperpłaszczyzną to posługujemy się klasyfikatorem wektorów nośnych. W odróżnieniu od klasyfikatora maksymalnego marginesu, klasyfikator wektorów nośnych nie wyznacza największego marginesu dla hiperpłaszczyzny separującej w sposób, by żadna z obserwacji nie znalazła się po stronie, do której nie należy. Natomiast pozwala na położenie niektórych obserwacji po niepoprawnej stronie hiperpłaszczyzny separującej. Umożliwia to dużo bardziej wydajną pracę klasyfikatora w przypadkach, dla których klasyfikator maksymalnego marginesu nie miałby szans powodzenia. Klasyfikacja dokonywana jest na podobnych zasadach co w (5). Jednakże w problemie optymalizacji klasyfikatora występują pewne parametry (*slack parameters*), które pozwalają niewielkiej liczbie przypadków na niepoprawną klasyfikację.

1.3 Maszyna wektorów nośnych

Maszyna wektorów nośnych jest rozszerzeniem koncepcji klasyfikatora wektorów nośnych poprzez zwiększenie przestrzeni cech z wykorzystaniem jąder (ang. *kernels*). Zagadnienia związane z SVM oraz z jądrami nie są trywialne zatem w tej sekcji przedstawimy ten problem tylko pobieżnie. SVM jest podejściem do klasyfikacji, w którym granica decyzyjna (hiperpłaszczyzna separująca) nie musi być liniowa. W przypadku, gdy liniowa granica decyzyjna nie spełnia swojego zadania, możliwe jest powiększenie przestrzeni cech o obserwacje podniesione np. do drugiej potęgi. W ten sposób rozwiązuje się problem nieliniowości. To rozwiązanie niesie jednak ze sobą niebezpieczeństwo powstania liczby danych, z którą komputerowi będzie bardzo trudno sobie poradzić. SVM pozwala na powiększenie przestrzeni cech w sposób najbardziej wydajny. Dokonuje tego za pomocą funkcji nazywanej jądrem, która jest miarą podobieństwa dwóch obserwacji (uogólnienie iloczynu skalarnego). Można powiedzieć także, że jądro to funkcja określająca w sposób ilościowy podobieństwo dwóch obserwacji. Przykładowy wzór

$$K(x_j, x_{j'}) = \sum_{i=1}^{d} x_j^{(i)} x_{j'}^{(i)}, \tag{7}$$

gdzie K jest jądrem, a x_j i $x_{j'}$ są badanymi obserwacjami. Jądro postaci (2.44) w rezultacie zastosowania do SVM daje nam klasyfikator wektorów nośnych. Jeśli jądro jest funkcją nieliniową mówimy o SVM.

2 Przykład

2.1 Opis zbioru

W przykładzie implementacji SVM posłużymy się zbiorem danych z pakietu *MASS* o nazwie *cats*. Zbiór zawiera informacje na temat masy serca i masy ciała kotów obu płci. Wszystkie badane koty były osobnikami dorosłymi i ważyły co najmniej 2 kg.

2.2 Cel

Naszym celem będzie zastosowanie maszyny wektorów nośnych do zaklasyfikowania płci danego kota na podstawie masy ciała i masy serca.

2.3 Kod programu

W pierwszym kroku ładujemy pakiety, z których będziemy korzystali

```
library(MASS)
library(e1071)
library(caret)
library(OneR)
library(BBmisc)
```

Pakiet *Mass*, tak jak wspomnieliśmy, zawiera zbiór *cats*. Sprawdzamy więcej informacji na temat naszego zbioru danych:

```
data(cats)
dane <- cats

str(dane)
summary(dane)
```

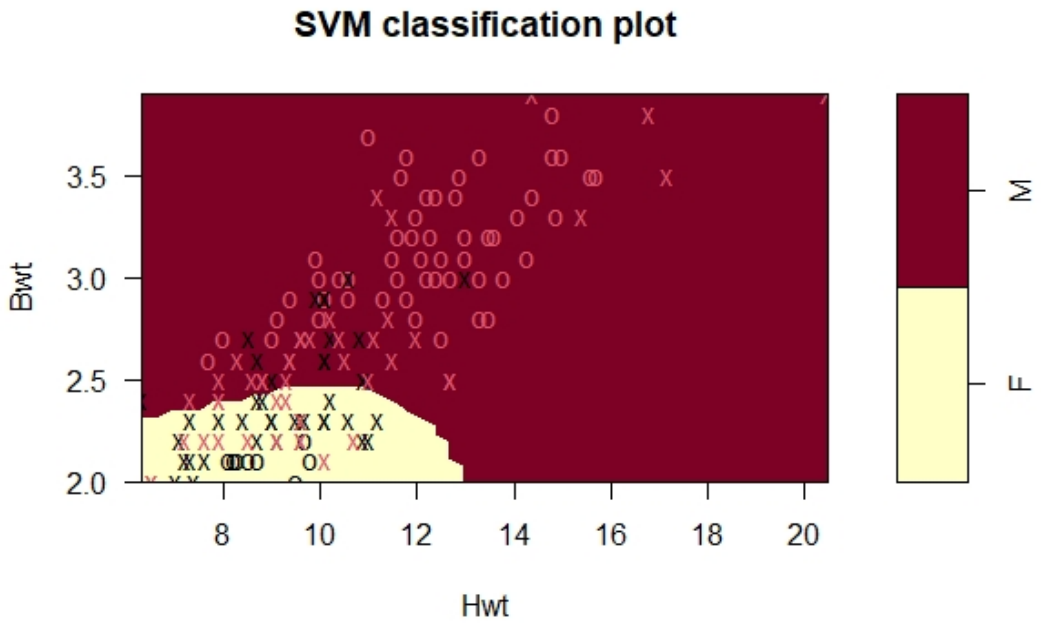
```
'data.frame': 144 obs. of  3 variables:
 $ Sex: Factor w/ 2 levels "F","M": 1 1 1 1 1 1 1 1 1 1 ...
 $ Bwt: num  2 2 2 2.1 2.1 2.1 2.1 2.1 2.1 2.1 ...
 $ Hwt: num  7 7.4 9.5 7.2 7.3 7.6 8.1 8.2 8.3 8.5 ...

  Sex          Bwt              Hwt
 F:47   Min.   :2.000   Min.   : 6.30
 M:97   1st Qu.:2.300   1st Qu.: 8.95
        Median :2.700   Median :10.10
        Mean   :2.724   Mean   :10.63
        3rd Qu.:3.025   3rd Qu.:12.12
        Max.   :3.900   Max.   :20.50
```

Zbiór zawiera 144 obserwacje o trzech cechach, kolejno:

- *Sex* – płeć danego kota (kategoryczna zmienna celu),
- *Bwt* – masa jego ciała (zmienna numeryczna),
- *Hwt* – masa jego serca (zmienna numeryczna).

W rozważanym zbiorze liczba samców to 97, a samic 47. Tworzymy model SVM dla całego zbioru w celu wizualizacji:

```
model_dane <- svm(Sex ~ ., data = dane)
print(model_dane)
```



Rysunek 3: Zastosowanie maszyny wektorów nośnych do zbioru *cats*.

Na rysunku 3 przedstawiona jest granica decyzyjna dla naszego przypadku. Żółta część oznacza strefę klasyfikacji nowej obserwacji jako osobnika płci żeńskiej, z kolei część czerwona jako osobnika płci męskiej.

Za pomocą funkcji *normalize* normalizujemy dane:

```
dane_norm <- normalize(dane, method = "range", range = c(0,1))
```

W celu kontroli wyświetlamy 6 początkowych i 6 końcowych wierszy oraz podsumowanie informacji o znormalizowanych danych

```
head(dane_norm)
tail(dane_norm)
```

```
  Sex         Bwt         Hwt
1   F 0.00000000 0.04929577
2   F 0.00000000 0.07746479
3   F 0.00000000 0.22535211
4   F 0.05263158 0.06338028
5   F 0.05263158 0.07042254
6   F 0.05263158 0.09154930

    Sex        Bwt       Hwt
139   M 0.8421053 0.6126761
140   M 0.8947368 0.3309859
141   M 0.9473684 0.5985915
142   M 0.9473684 0.7394366
143   M 1.0000000 0.5704225
144   M 1.0000000 1.0000000
```

Zwróćmy uwagę, że pojawiły się wartości zerowe. Nie jest to niewłaściwe dlatego, że wartości najmniejsze tzn. wagę 2 kg zastąpiono wartością 0.

Dokonujemy podziału zbioru na zbiór treningowy i zbiór testowy w stosunku 3:2, a następnie tworzymy model maszyny wektorów nośnych:

```
podzial <- createDataPartition(dane_norm$Sex, p = 0.60, list = FALSE)
trening <- dane[podzial, ]
test <- dane[-podzial, ]
model_trening <- svm(Sex ~ ., data = trening)
```

Wykonujemy predykcję i wyświetlamy informacje na temat modelu:

```
predykcja <- predict(model_trening, test[, -1], type = "class")
eval_model(predykcja, test[,1])
```

```
Confusion matrix (absolute):
          Actual
Prediction  F  M Sum
       F    8  6  14
       M   10 32  42
       Sum 18 38  56

Confusion matrix (relative):
          Actual
Prediction    F    M  Sum
       F   0.14 0.11 0.25
       M   0.18 0.57 0.75
       Sum 0.32 0.68 1.00

Accuracy:
0.7143 (40/56)

Error rate:
0.2857 (16/56)

Error rate reduction (vs. base rate):
0.1111 (p-value = 0.3392)
```

Nasz model dokonał poprawnej predykcji płci kota 40 razy na 56 badane przypadki, co daje dokładność na poziomie 71.43%. 6 osobników płci męskiej zaklasyfikowano jako osobniki płci żeńskiej i 10 osobników płci żeńskiej zaklasyfikowano jako osobniki płci męskiej.