Praca domowa 3 - Wstęp do uczenia maszynowego

Oliwia Trzcińska

1 Wstęp

Celem pracy domowej jest implementacja oraz przetestowanie metody k najbliższych sąsiadów.

2 Implementacja

Do stosowania metody k najbliższych sąsiadów służy funkcja `knn()`. Przyjmuje ona jako argumenty kolejno:

- zbiór treningowy - macierz rzeczywistą $\mathbb{X}$ rozmiaru $n \times c$, która reprezentuje n punktów w $\mathbb{R}^c$,
- y - n - elementowy obiekt taki, że y_i to etykieta odpowiadająca obserwacji $\mathbb{X}[i,]$,
- zbiór testowy - macierz rzeczywistą $\mathbb{Z}$ rozmiaru $m \times c$, która reprezentuje m punktów w $\mathbb{R}^c$,
- k - liczbę całkowitą z przedziału $[1, n]$, która oznacza liczbę najbliższych sąsiadów biorących udział w poszukiwaniu etykiety odpowiadającej punktom ze zbioru testowego,
- p - wartość rzeczywistą nie mniejszą od 1 lub wartość `'infty'` określającą, która metryka Minkowskiego L_p będzie używana do poszukiwania najbliższych sąsiadów (domyślna wartość p to 2).

Funkcja zwraca m-elementowy obiekt w taki, że w_i to etykieta odpowiadająca obserwacji $\mathbb{Z}[i,]$. Funkcja `knn()` działa przy założeniu, że w wejściowych zbiorach treningowym i testowym nie ma braków danych oraz dane są odpowiednio przeskalowane. Korzysta z funkcji pomocniczych:

- `check_params()`, która sprawdza zgodność wprowadzonych parametrów z powyższą definicją,
- `lp_distance(a, b, p)`, która oblicza odległość między wektorami a i b w metryce L_p.

Mechanizm działania funkcji `knn()` jest następujący:

1. sprawdzamy poprawność argumentów wejściowych, w przypadku jakichkolwiek niezgodności kończymy działanie funkcji
2. tworzymy pustą listę w, do której będziemy później zapisywać przewidywane etykiety
3. dla $i = 1, 2, 3, \ldots m$ tworzymy listę odległości i-tego wiersza zbioru testowego od każdego z wierszy zbioru treningowego $\mathbb{X}$ w metryce L_p
4. dla danej obserwacji ze zbioru testowego znajdujemy indeksy k najbliższych obserwacji ze zbioru treningowego
5. wyznaczamy ciąg etykiet k najbliższych sąsiadów tej obserwacji i szukamy wartości najczęściej występującej (jeśli jest ich kilka, losowo wybieramy jedną, każdą z tym samym prawdopodobieństwem)
6. zapisujemy przewidywaną etykietę do listy w
7. po przeprowadzeniu tej procedury dla każdej obserwacji ze zbioru $\mathbb{Z}$ zwracamy listę w.

3 Testy

3.1 Test 1

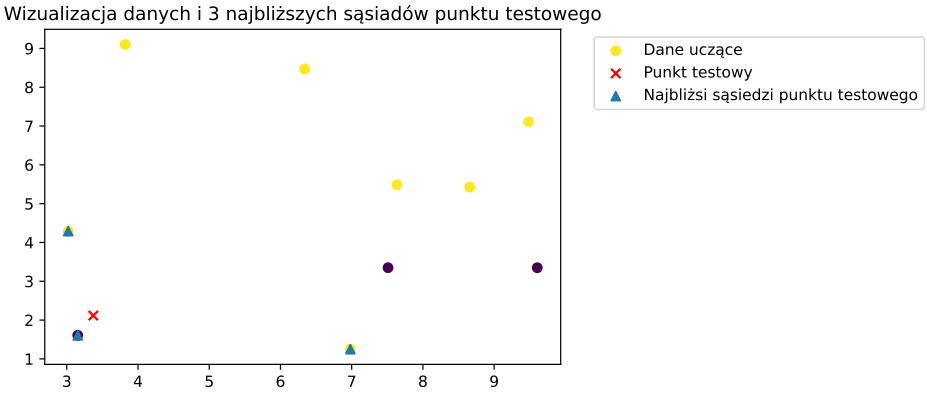
Najpierw sprawdzimy działanie funkcji `knn()` w przypadku, gdy $k = 1$ oraz zbiory treningowy i testowy są takie same. Przy pomocy funkcji `make_classification()` generujemy losowy problem klasyfikacji. Liczba naszych obserwacji to 50, a macierz $\mathbb{X}$ ma 2 kolumny. Zmieniamy typ danych na listy i wywołujemy trzy razy funkcję `knn()`, kolejno z parametrami $p = 1, 2$, `'infty'`. Okazuje się, że przy takich założeniach za każdym razem jest zwrócony idealnie odtworzony wektor prawdziwych etykiet.

3.2 Test 2

W tym eksperymencie patrząc na wykres danych sprawdzimy, czy funkcja `knn()` poprawnie przewidzi etykietę. Generujemy zbiór treningowy losując z rozkładu jednostajnego 10 punktów w dwuwymiarowej przestrzeni, o współrzędnych z przedziału $(0, 10)$. Następnie każdemu z nich losowo wybieramy etykietę 0 lub 1, każdą z prawdopodobieństwem 0,5. Ustalamy także losowo jeden punkt testowy, dla którego etykietę będziemy chcieli przewidzieć przy pomocy metody 3 najbliższych sąsiadów. Do obliczania odległości stosujemy metrykę L_2.

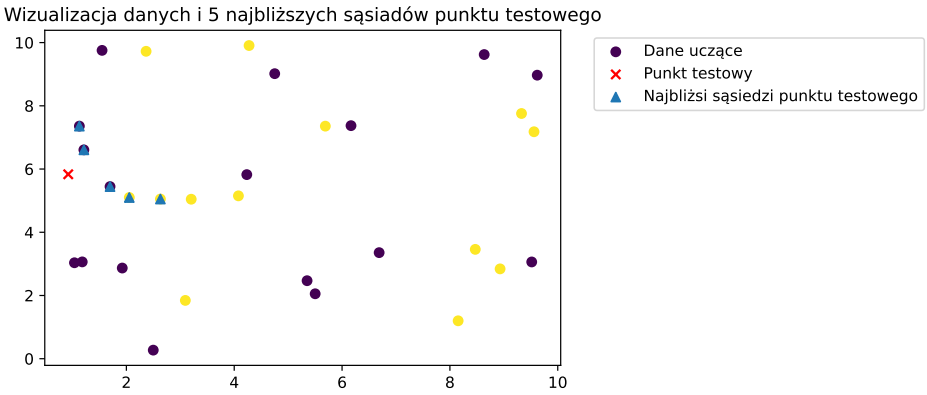
Na wykresie kropkami są zaznaczone obserwacje ze zbioru treningowego. Odpowiednio kolorem żółtym te, które mają etykietę 1 i fioletowym te, które mają etykietę 0. Czerwony krzyżyk oznacza punkt testowy. Korzystając z wcześniej opisanej funkcji `lp_distance()` obliczamy odległości punktu testowego od każdej obserwacji ze zbioru treningowego, wybieramy 3 najbliższe punkty i oznaczamy je trójkącikiem.

Widzimy, że dwoje najbliższych sąsiadów jest oznaczonych kolorem żółtym, a jeden fioletowym. Funkcja `knn()` poprawnie więc zaklasyfikowała obserwację testową do klasy 1.



Przeprowadzimy teraz analogiczny eksperyment, tym razem dla $k = 5$. Zbiorem treningowym będzie losowo (z rozkładu jednostajnego) wygenerowane 30 punktów w dwuwymiarowej przestrzeni, o współrzędnych z przedziału $(0, 10)$. Ponownie kolorem fioletowym oznaczone są obserwacje z etykietą 0.

Trzy spośród pięciu najbliższych punktów mają etykietę 0, taka jest też wartość zwrócona przez naszą funkcję.



3.3 Test 3

Sprawdzimy teraz jaka jest dokładność zaimplementowanej metody k najbliższych sąsiadów. Ponownie generujemy losowy problem klasyfikacji. Tym razem mamy 1000 obserwacji 2 cech. Dzielimy dane na zbiory treningowy i testowy w proporcji 7:3. Sprawdzamy dokładność na obu tych zbiorach w zależności od parametrów `k` oraz `p`. Rozpatrujemy $k \in \{1, 3, 5, 7\}$ oraz metryki L_1, L_2 i L_∞. Wyniki zostały przedstawione w poniższej tabeli.

Parametry	Wartość na zbiorze treningowym	Wartość na zbiorze testowym
$k = 1, L_1$	1,000	0,957
$k = 1, L_2$	1,000	0,957
$k = 1, L_\infty$	1,000	0,953
$k = 3, L_1$	0,964	0,970
$k = 3, L_2$	0,966	0,973
$k = 3, L_\infty$	0,963	0,973
$k = 5, L_1$	0,957	0,967
$k = 5, L_2$	0,959	0,970
$k = 5, L_\infty$	0,957	0,973
$k = 7, L_1$	0,966	0,970
$k = 7, L_2$	0,964	0,967
$k = 7, L_\infty$	0,964	0,973

Przeprowadzimy ten sam test jeszcze raz, teraz na mniejszym zbiorze danych. Analogicznie generujemy problem klasyfikacji z 50 obserwacjami, dzielimy dane na zbiory treningowy i testowy w proporcji 7:3. Sprawdzamy działanie funkcji `knn()` dla tych samych parametrów co wcześniej. Wyniki prezentują się następująco.

Parametry	Wartość na zbiorze treningowym	Wartość na zbiorze testowym
$k = 1, L_1$	1,000	0,867
$k = 1, L_2$	1,000	0,867
$k = 1, L_\infty$	1,000	0,933
$k = 3, L_1$	0,943	0,933
$k = 3, L_2$	0,943	0,933
$k = 3, L_\infty$	0,971	0,933
$k = 5, L_1$	0,914	0,867
$k = 5, L_2$	0,886	0,867
$k = 5, L_\infty$	0,857	0,867
$k = 7, L_1$	0,829	0,867
$k = 7, L_2$	0,886	0,867
$k = 7, L_\infty$	0,886	0,800

4 Podsumowanie

Zaimplementowana metoda k najbliższych sąsiadów osiągnęła dobre wyniki w problemie klasyfikacji na wygenerowanych przez nas danych. Nawet przy małym zbiorze treningowym jakość predykcyjna okazała się zadowalająca.