Wprowadzenie do sztucznej inteligencji - Laboratorium 7

Ireneusz Okniński 310228

Styczeń 2022

1 Wstęp

1.1 Treść zadania

Zaimplementuj naiwny klasyfikator Bayesa oraz zbadaj działanie algorytmu w zastosowaniu do zbioru danych Iris Data Set. Pamiętaj, aby podzielić zbiór danych na zbiór trenujący oraz uczący!

1.2 Technologia

Zadanie zostało zaimplementowane w języku Python z wykorzystaniem bibliotek numpy, scikit-learn, matplotlib.

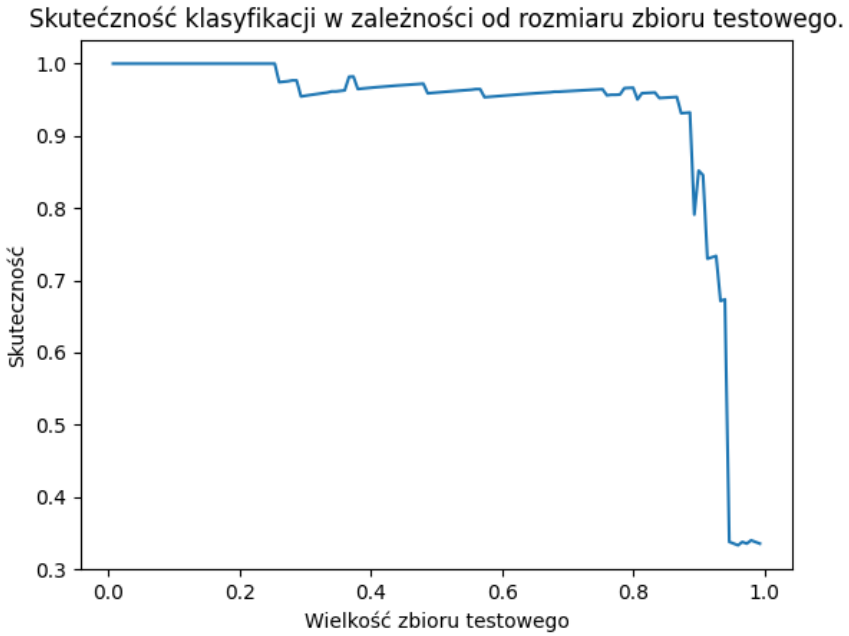
1.3 Zbiór danych

Dane składają się z 4 atrybutów: "sepal length", "sepal width", "petal length", "petal width", oraz przydzielonej klasy. Wynikowe klasy to: "Iris-setosa", "Iris-versicolor", "Iris-virginica". Kolejne klasy w zbiorze danych występują bezpośrednio po sobie. Dlatego też, aby uczenie modelu miało sens należy pomieszać zbiór danych.

2 Eksperymenty

Naiwny klasyfikator Bayesa nie posiada hiperparametrów, których wartość regulowałaby jego działanie. Zbadałem zatem wpływ podziału zbioru danych na uczący i testowy.

2.1 Podział danych na zbiór uczący i testowy

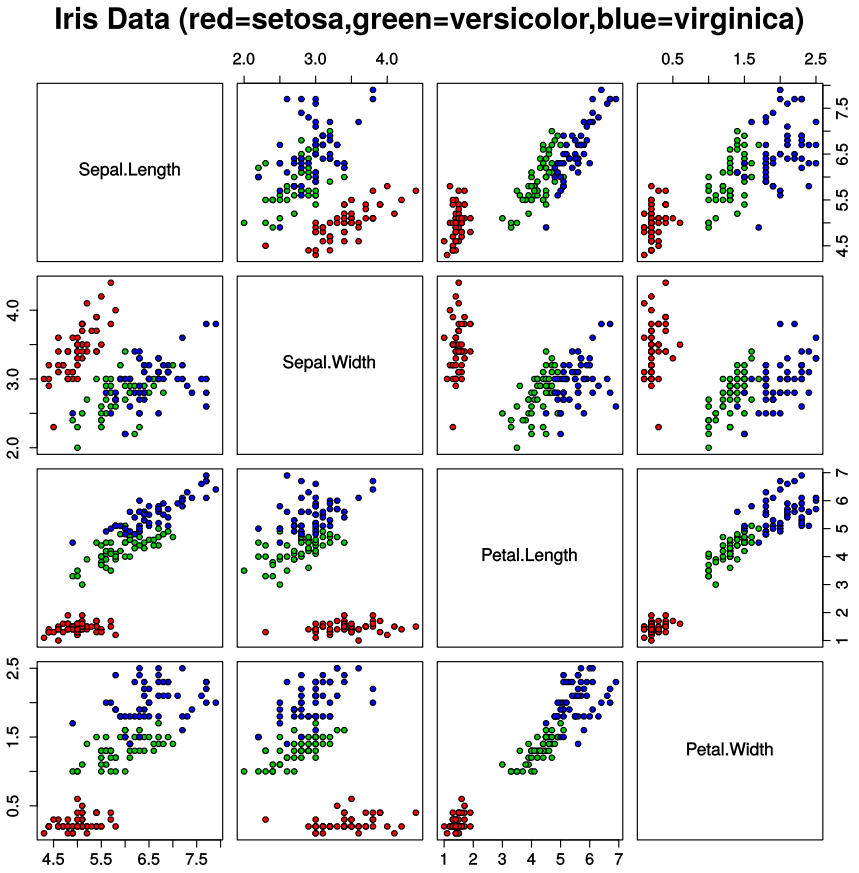


Rysunek 1: Wykres zależności skuteczności od rozmiaru zbioru testowego

Klasyfikator osiągał zadowalającą skuteczność, gdy stosunek zbiorów testowego do uczącego był mniejszy od ok. 85:15. Gdy stosunek ten był większy od ok. 85:15 skuteczność algorytmu bardzo spadała, aż do maksymalnej wartości ok 1/3, co świadczy o tym, że skuteczność klasyfikacji była bliska losowej.

3 Wnioski

Naiwny klasyfikator Bayesa potrafi zbudować skuteczny model nawet przy małej ilości danych treningowych. Należy zauważyć, że problem klasyfikacji irysów był prosty, przy bardziej złożonych problemach model ten mógłby nie być już tak skuteczny.



Rysunek 2: Wykresy zbioru Iris data set. https://en.wikipedia.org/wiki/Iris$_flower_data_set$

Widoczne jest, że dla niektórych parametrów klasy są liniowo separowalne. Może to być jedną z przyczyn tak wysokiej skuteczności algorytmu.