Отчет по лабораторной работе №3

**Доп. задание** №**2**

*Аксенова Валерия, Коваленко Александр, Шустров Андрей*

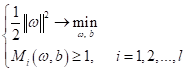
***Постановка задачи:***

Рассмотрим постановку задачи оптимизации в методе обучения *support vector* machine ( метод опорных векторов):

Для классификации бинарных образов мы хотим провести такую гиперплоскость, которая будет корректно отделять один класс от другого, ориентируясь только на распределение обучающей выборки и по возможности без дополнительных предположений о распределении образов в классах.

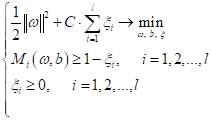
С точки зрения *SVM*, *оптимальная разделяющая гиперплоскость* – это та, которая образует наиболее широкую полосу между объектами двух классов. При этом сама разделяющая гиперплоскость будет точно проходить посередине этой полосы.

*Задача оптимизации для общего случая:*



*Верхнее выражение определяет ширину полосы, нижнее - расстояние от разделяющей гиперплоскости до выбранного образа ( margin).*

В случае нелинейного разделения вводим *slack variables* - некоторый штраф за нарушение исходного неравенства:



*, где C – гиперпараметр, определяющий степень минимизации величин* *.*

***Описание методов:***

1. *prepare\_data(x)* ***-*** добавляет столбец единиц (*bias*) к каждому элементу входных данных x. Это необходимо для корректного обучения модели.
2. *train\_svm(x\_train, y\_train) -* обучает линейную модель *SVM* с использованием библиотеки *sklearn.* Возвращает обученный классификатор.
3. *plot\_svm(x\_train, y\_train, support\_vectors, line\_coords=None)* - строит график данных, включая точки классов, опорные векторы и (при наличии) разделяющую линию. Использует библиотеку *matplotlib*.

Для *линейно разделимого случая* возьмем датасет:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| *№* | *Width* | *Length* | *Class* | |
| *1* | *10* | *50* | *blue* | *-1* |
| *2* | *20* | *30* | *red* | *+1* |
| *3* | *25* | *30* | *red* | *+1* |
| *4* | *20* | *60* | *blue* | *-1* |
| *5* | *15* | *70* | *blue* | *-1* |
| *6* | *40* | *40* | *red* | *+1* |
| *7* | *30* | *45* | *red* | *+1* |
| *8* | *20* | *45* | *blue* | *-1* |
| *9* | *40* | *30* | *red* | *+1* |
| *10* | *7* | *35* | *blue* | *-1* |

Для *линейно не разделимого случая* добавим в существующий датасет:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| *№* | *Width* | *Length* | *Class* | |
| *11* | *30* | *10* | *blue* | *-1* |
| *12* | *15* | *50* | *red* | *+1* |

и будем использовать класс SVC, так как прежний LinearSVC не применим к новой выборке.

***Результаты:***

После запуска программы *Linear SVM* увидим следующие значения:

- *коэффициентов вектора ω:*

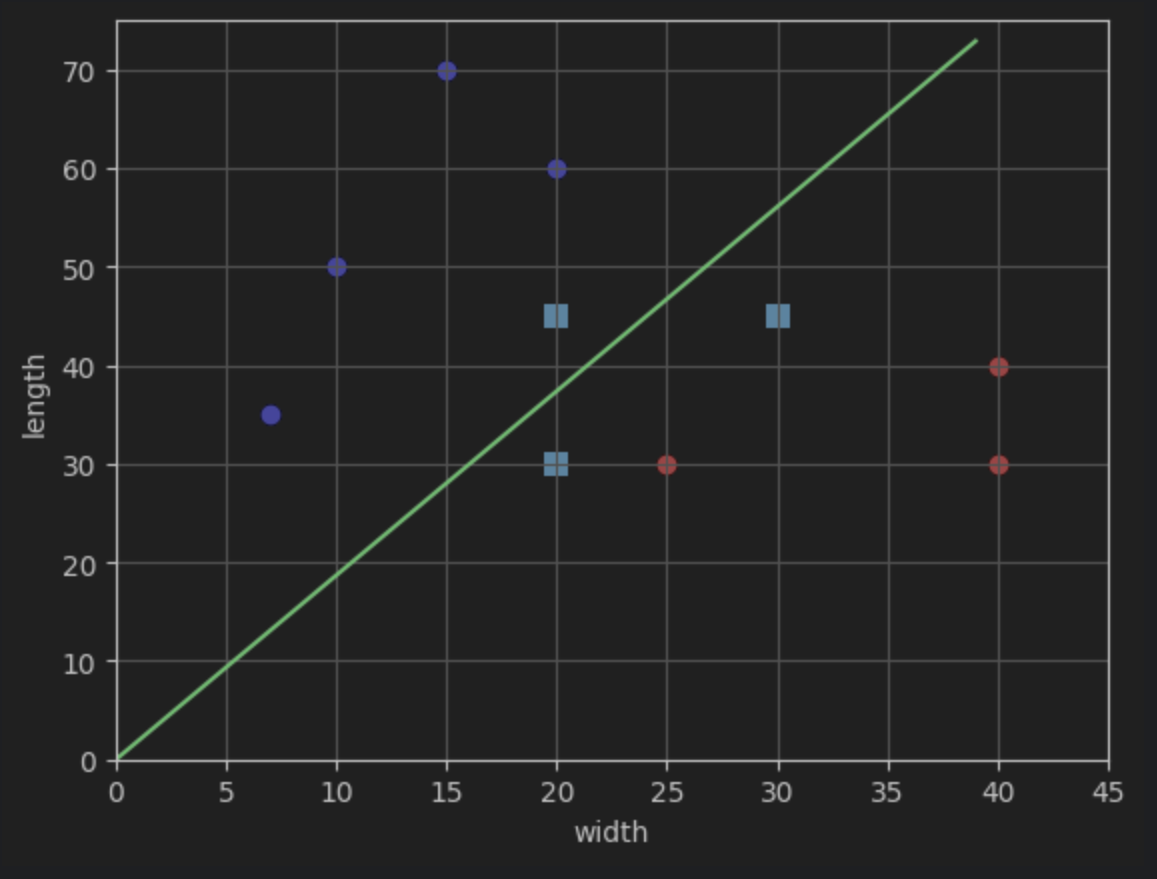
[ 0.24371833 -0.13071248 0.01218592]

- *список опорных векторов, для которых :*

[[20. 45. 1.]

[20. 30. 1.]

[30. 45. 1.]]



Разделяющая линия действительно проходит по центру полосы, образованной опорными векторами.

После запуска *Nonlinear SVM* мы увидим *качество классификации*:

[-2 2 0 0 0 0 0 0 0 0 0 0]

( нули соответствуют верной классификации). Кроме того, видим *список опорных векторов*:

[[30. 10. 1.]

[20. 60. 1.]

[20. 45. 1.]

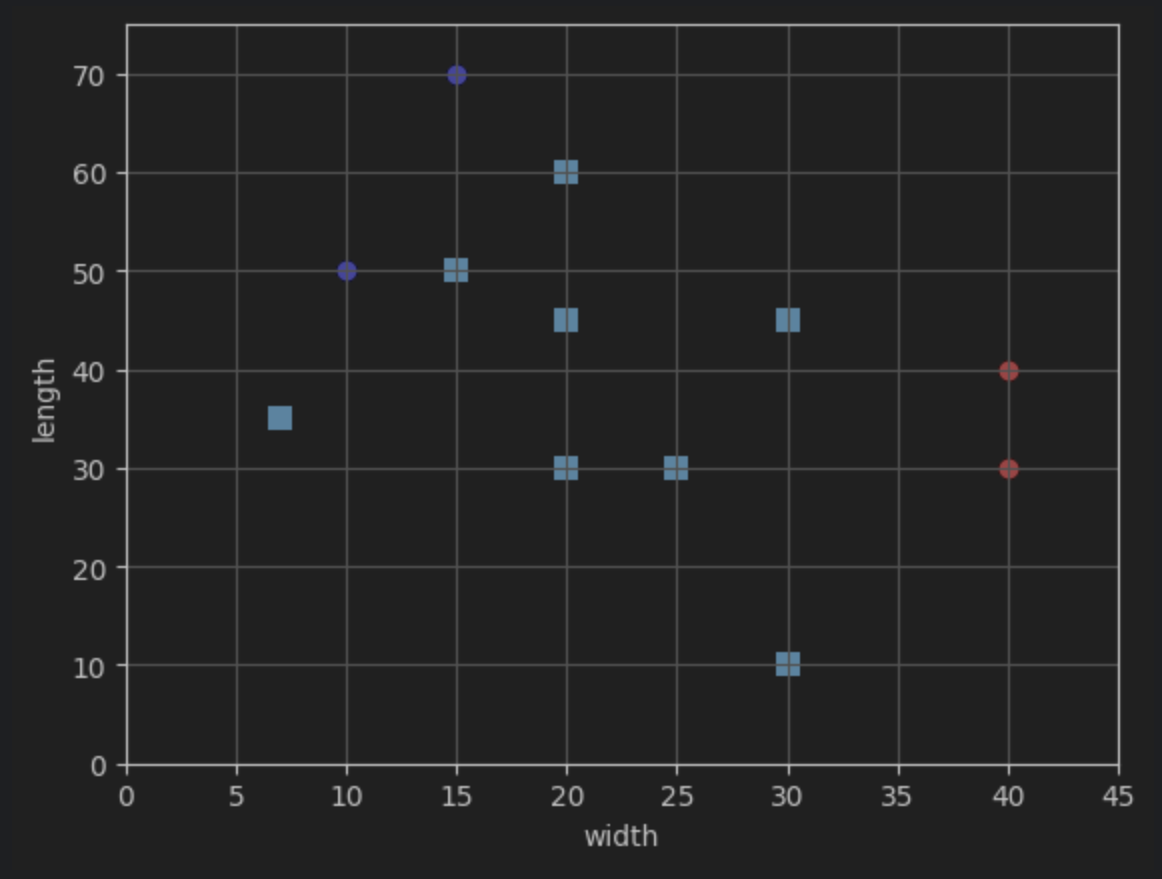
[ 7. 35. 1.]

[15. 50. 1.]

[20. 30. 1.]

[25. 30. 1.]

[30. 45. 1.]]

и их стало заметно больше предыдущего случая (при линейно разделимой выКлассификатор ошибся только на двух первых наблюдениях, в которые мы прописали выбросы, то есть, он корректно построил разделяющую гиперплоскость.