Мартынов Д.А.

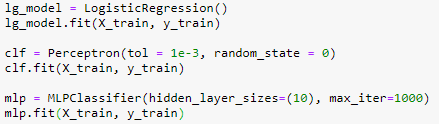
МПИ-20-4-2

Лабораторная работа 1

В работе производится сравнение линейного классификатора и однослойного перцептрона на двух датасетах. В качестве первого датасета были выбраны Ирисы Фишера, а второй датасет генерировался автоматически со следующим набором параметров: *количество элементов, количество признаков, количество классов, стандартное отклонение*. Далее датасет разбивается на обучающую, валидационную и тестовую выборку: берется случайное число *n* из указанного диапазона, первые *n* элементов относятся к обучающей выборке, следующие *n* элементов к валидационной, остальные к тестовой. Диапазон зависит от количества элементов в датасете. Набор параметров, использованный в работе: *400, 2, 4, 2*, *диапазон (136, 156)*.

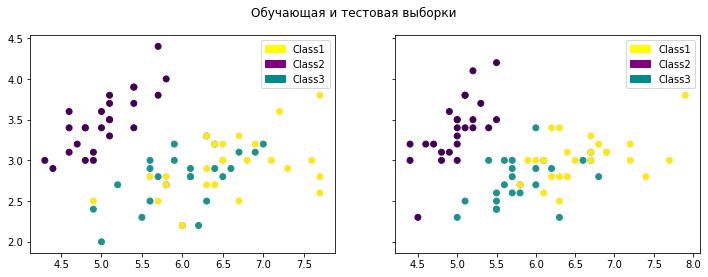
Были взяты две реализации однослойного перцептрона из библиотеки skitlearn, так как результаты на одинаковых датасетах получились разные.

Первый перцептрон – sklearn.Perceptron с параметром tol = 10-3, второй – sklearn.MLPClassifier, который создавался с одним слоем из 10 нейронов (hidden\_layer\_sizes=(10)) и ограничением на максимальное количество итераций в 10000. Функция активации во втором случае – ReLu, функция оптимизации Adam (стоит по умолчанию), tol = 10-3.

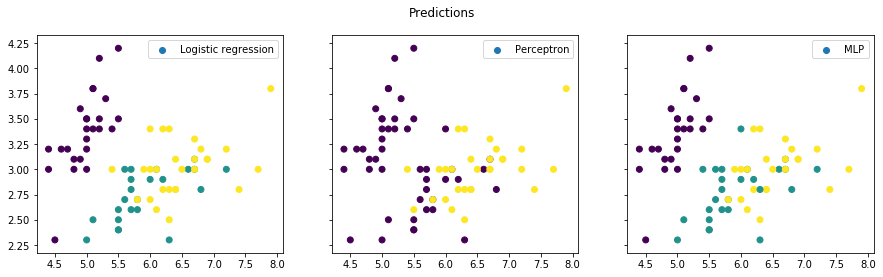
****

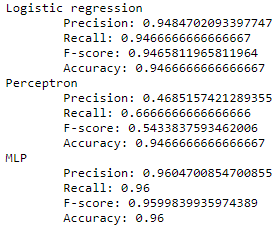
**Датасет Ирисы Фишера**

На графиках обучающая и тестовая выборки (их двумерная проекция):



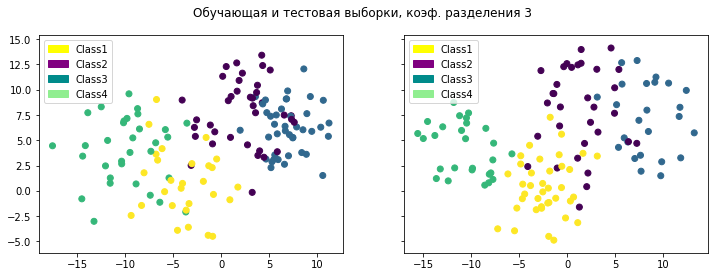
И результаты классификации с метриками (также в виде двумерной проекции):



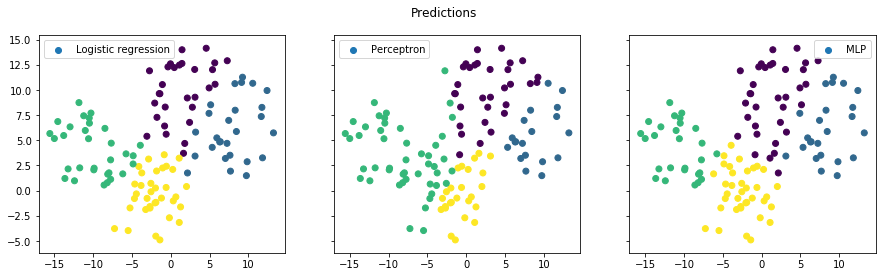


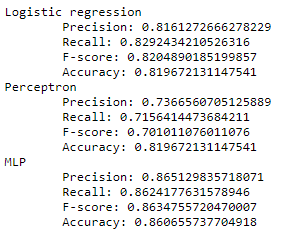
**Генерируемый датасет**

Обучающая и тестовая выборки:

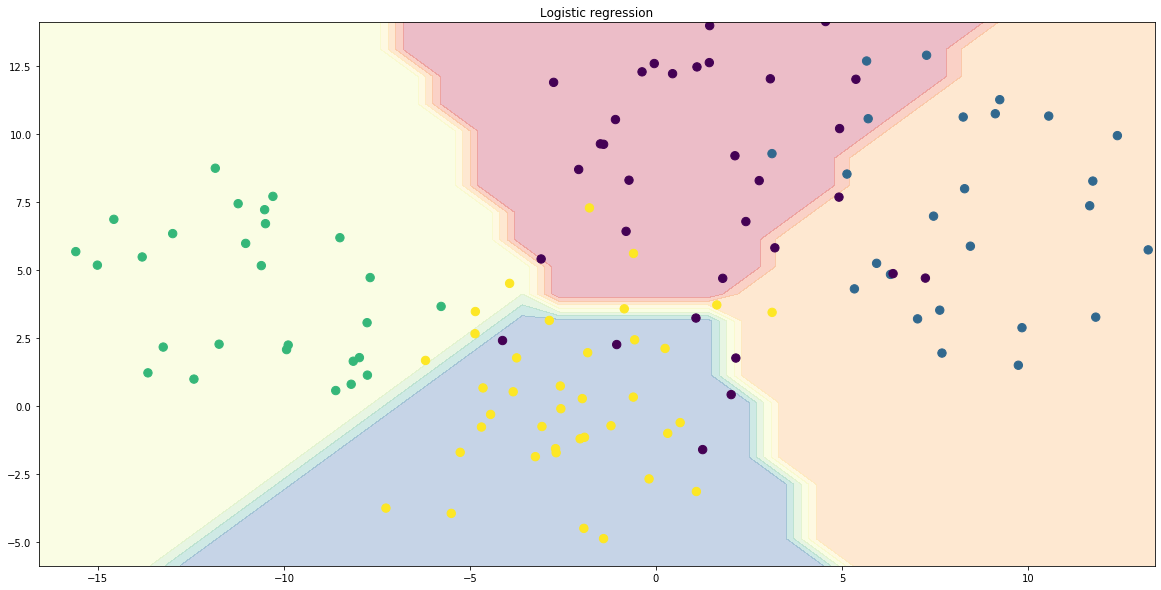


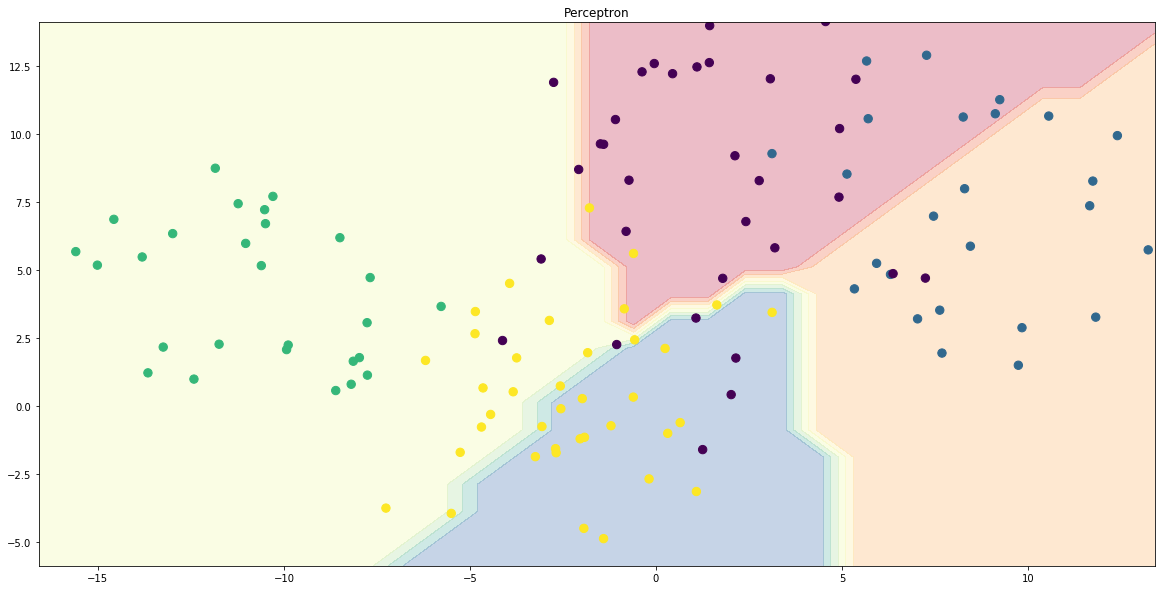
Результаты классификации на тестовой выборке и метрики:

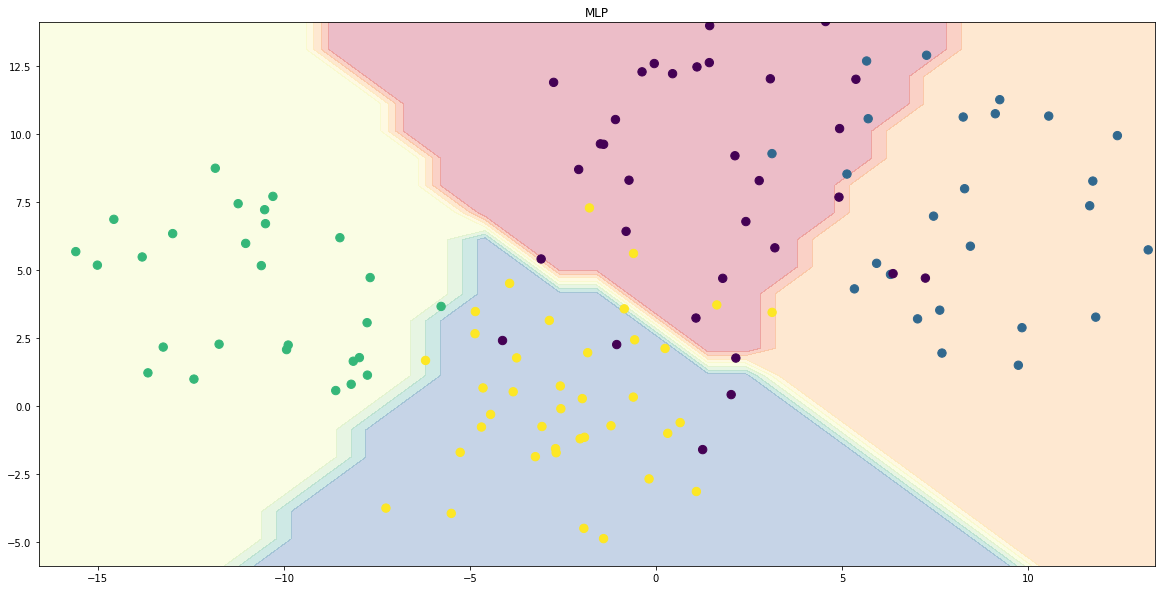




На следующих графиках показана визуализация разделения. Истинная принадлежность классам отображена цветами точек, а заливкой показаны классы, определенные классификаторами:





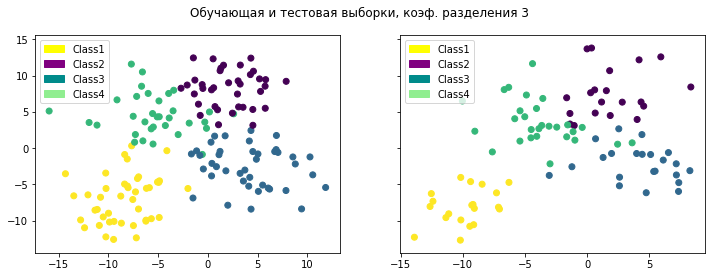


В обоих случаях перцептрон, реализованный в sklearn.Perceptron, показал себя хуже двух других классификаторов, а второй перцептрон имеет самые высокие оценки метрик качества классификации.

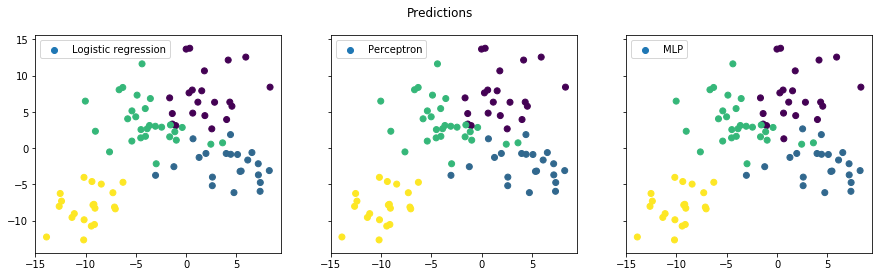
Исходя из этих двух сценариев, можно сказать, что однослойный перцептрон, реализованный в sklearn.MLPClassifier, лучше других подходит для выбранной задачи классификации.

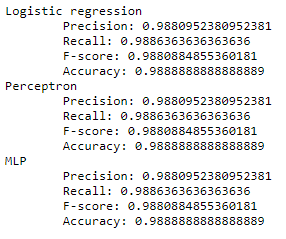
Теперь поменяем параметры генерации датасета, добавив еще 3 признака (остальные параметры прежние)

Обучающая и тестовая выборки (двумерная проекция):



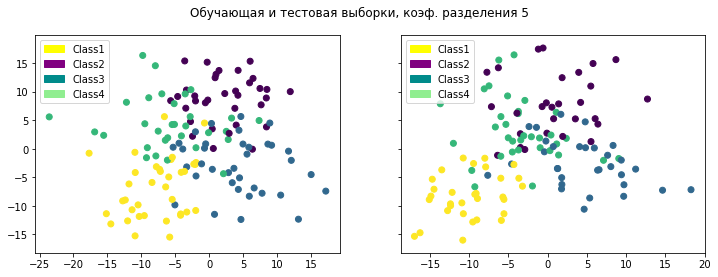
Результаты классификации (двумерная проекция) и метрики:

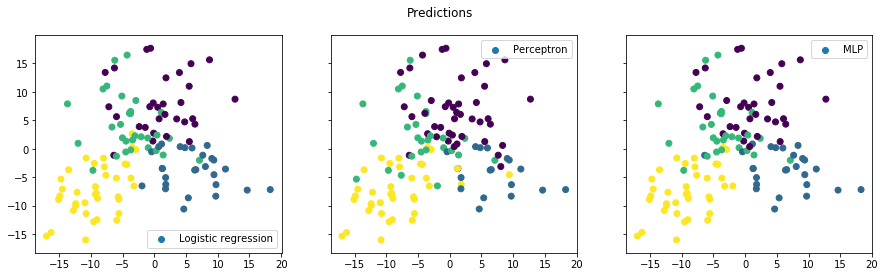


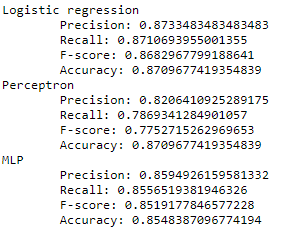


После увеличения количества признаков все классификаторы показывают одинаковый, одинаково хороший, результат. Поэтому в случае большого количества признаков стоит использовать тот классификатор (из рассмотренных в данной работе), который работает быстрее.

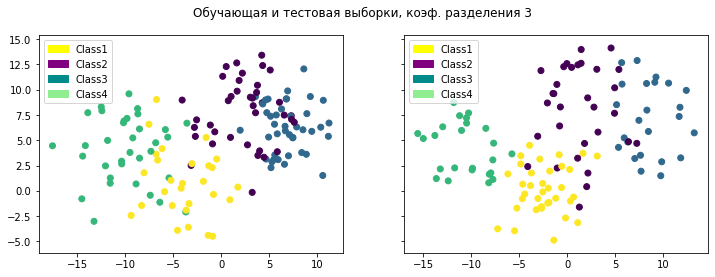
Теперь попробуем увеличить степень пересечения классов, подняв стандартное отклонение до 5:

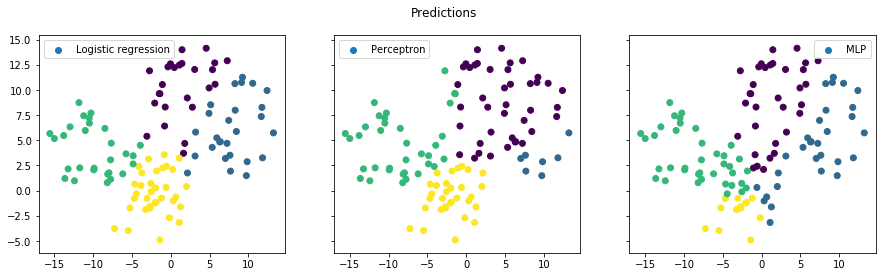


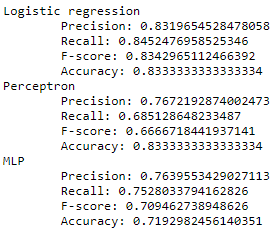




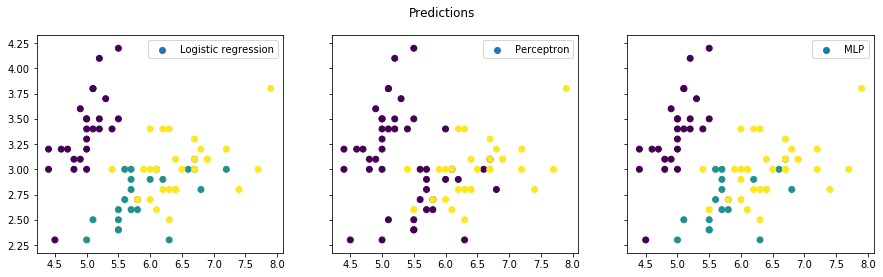
Если вернуться к варианту датасета с 2 признаками и дать второму перцептрону всего 4 нейрона (hidden\_layer\_sizes=(4)), равное количеству классов, то качество классификации сильно упадет по сравнению с вариантом с 10 нейронами

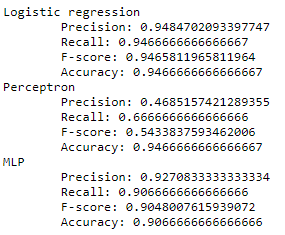






Хотя на датасете Ирисы Фишера результат ухудшился не сильно





**Сравнительная таблица**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | Logistic regression | Sklearn.Perceptron | Sklearn.MLP  10 neurons | Sklearn.MLP  4 neurons |
| Iris |  |  |  |  |
| Generated dataset | | | | |
| Claster\_std 3 |  |  |  |  |
| Claster\_std 5 |  |  |  |  |
| Claster\_std 3, number of features 5 |  |  |  |  |

**Выводы**

Исходя из результатов последних трех примеров, можно сказать, что однослойный перцептрон может показывать результаты лучше линейного классификатора в том случае, если в его единственном слое достаточно нейронов (желательно больше, чем количество классов). Если требуется использовать именно однослойный перцептрон, то я бы выбрал реализацию MLP, выставив в параметрах один слой, вместо Perceptron.