Бинарная классификация (линейные модели)

Практическое задание для самостоятельного выполнения

Задание 1.

1. Используя модуль ***datasets*** библиотеки ***Scikit-learn***, сгенерировать модельный набор данных для задачи бинарной классификации в виде двух облаков точек: общее количество точек равно 200, количество признаков, характеризующих объекты, равно 2. Параметр, определяющий степень рассеянности данных, установить равным 5.5. Обеспечить воспроизводимость результатов, задав значение соответствующему параметру.
2. Вывести сгенерированные координаты точек и метки классов.
3. Выполнить визуализацию сгенерированных облаков.
   Указание: для представления объектов разных классов точками разных цветов можно использовать объект ***ListedColormap*** (инструментарий библиотеки ***Matplotlib***):

   ```
   from matplotlib.colors import ListedColormap
   colors = ListedColormap(['red', 'blue'])   # список используемых цветов
   ```

   После этого объект с именем *colors* можно использовать в качестве значения параметра *cmap* в ***pylab.scatter***.
4. Поэкспериментировать с величиной шума: задать значения соответствующего параметра равными 4.0, 7.0 и 10.0, вывести графики в одном ряду с заголовками, сообщающими об используемом значении параметра шума.

Задание 2.

1. Выполнить разбиение набора данных, полученного в п. 1 задания 1, на обучающую и тестовую выборки в соотношении 70/30.
2. Создать модель линейной классификации, использующую $\boldsymbol{L_2}$-регуляризатор, и обучить ее на обучающей выборке (значение коэффициента регуляризации оставить по умолчанию).
3. Получить предсказания обученной модели для объектов тестовой выборки. Вывести массив ответов на тестовой выборке и массив предсказанных моделью значений. Оценить качество классификации с помощью метрики ***accuracy***; дать интерпретацию полученной оценки.
4. Создать несколько моделей линейной классификации, использующих $\boldsymbol{L_2}$- и $\boldsymbol{L_1}$-регуляризаторы и различные функции потерь, используя ***SGDClassifier***. Обучить построенные модели на обучающей выборке. Оценить качество всех полученных классификаторов.
5. Создать отчет по результатам выполнения п. 1: постановка задачи, описание каждой модели (используемая функция потерь, используемый регуляризатор, используемое значение коэффициента регуляризации, полученные результаты, выводы).

Задание 3.

1. Выбрать две лучшие (по метрике ***accuracy***) модели из числа классификаторов,

полученных при выполнении задания 2. Используя инструментарий модуля ***sklearn.metrics***, оценить качество этих моделей с помощью метрик ***precision***, ***recall*** и ***F–меры.***

2. Получить матрицу ошибок. Используя эту матрицу, посчитать (по формулам) значения ***precision***, ***recall*** и ***F–меры***, сравнить полученные значения с результатами, полученными в п. 1.
3. Получить для рассматриваемых моделей значения ***FPR*** и ***TPR*** (на обучающей и тестовой выборке отдельно). Выполнить вычисления с помощью функции ***roc_curve*** и непосредственно по матрице ошибок. Сравнить результаты.
4. Построить ROC-кривые (для обучающей и тестовой выборки).
5. Проанализировать все полученные результаты, дать им интерпретацию.