Тема 12. Линейные модели классификации

Практические задания для самостоятельного выполнения

Задания выполняется по вариантам. Формулировка задания общая для всех вариантов; конкретные условия, указанные в общей формулировке, выбираются согласно номеру вашего варианта и описанию задачи.

Результаты выполнения задания необходимо представить в виде двух файлов:

1) ноутбук в формате *ipynb*, содержащий программный код, результаты его выполнения, а также все необходимые пояснения, выводы и комментарии (в текстовых ячейках);
2) файл в формате *pdf*, полученный путем экспорта (или вывода на печать) ноутбука из п. 1).

Внимание: в названии файлов должна обязательно присутствовать фамилия автора и номер варианта. Безымянные работы проверяться не будут.

Задание 1 (максимум 1 балл)

Используя инструментарий библиотеки *sklearn*, реализовать вычислительные эксперименты с построением моделей линейной классификации, проанализировать их результаты и сделать выводы. В процессе работы выполнить следующие действия:

1. Сгенерировать модельный набор данных для задачи бинарной классификации по двум признакам в соответствии с номером своего варианта. Обеспечить воспроизводимость результатов, установив *random_state*, равный номеру своего варианта.
2. Вывести сгенерированные координаты точек и метки классов.
3. Выполнить визуализацию сгенерированного набора данных.
4. Выполнить разовое разбиение набора данных, полученного в п.2, на обучающую и тестовую выборки в соотношении 80/20, установив *random_state*, равный номеру своего варианта.
5. Создать модель линейной классификации, использующую L_2-регуляризатор, и обучить ее на обучающей выборке (значение коэффициента регуляризации оставить по умолчанию).

6. Вывести значения весов и свободного коэффициента, записать формулу полученного линейного классификатора.
7. Выполнить визуализацию разделяющей прямой линейного классификатора.
8. 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* на обучающей и тестовой выборке; дать интерпретацию полученным результатам.
9. Создать не менее восьми моделей линейной классификации, использующих L_2- и L_1-регуляризаторы с разными значениями коэффициента и различными функциями потерь, используя *SGDClassifier* (*random_state* задать равным номеру своего варианта). Обучить модели на обучающей выборке.
10. Оценить качество всех полученных классификаторов на обучающей и тестовой выборке.
11. Создать отчет по результатам выполнения пп. 8-10: описание каждой модели (используемая функция потерь, используемый регуляризатор, используемое значение коэффициента регуляризации), полученные результаты, выводы.

Задание 2 (максимум 0,5 балла)

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 (на обучающей и тестовой выборке отдельно).
2. Получить матрицу ошибок для тестовой выборки для каждой модели. Используя эти матрицы, посчитать (по формулам) значения *accuracy*, *precision*, *recall* и *F*–меры, сравнить полученные значения с результатами, полученными в п. 1.
3. Проанализировать все полученные результаты, дать им интерпретацию. Выбрать лучшую модель.

Задание 3 (максимум 1,5 балла)

В этом задании используется набор данных с результатами наблюдений за космосом, сделанных SDSS.

Описание признаков исходного набора данных.

Наборы данных по вариантам представлены в *csv*-файлах; они имеют одну и ту же структуру, соответствующую общему описанию, но отличаются набором записей. Имена файлов: *Вариант N.csv*, где *N* – номер варианта.

Выполнить первичное изучение имеющихся данных и построить линейную модель многоклассовой классификации. В процессе работы выполнить следующие действия:

1. Импортировать данные из файла и вывести несколько первых записей (для контроля корректности импорта и получения представления о наборе).
2. В рамках первичного знакомства с данными:
   - ознакомится с описанием признаков по ссылке выше;
   - изучить признаки на наличие пропущенных значений;
   - определить число классов и количество объектов по каждому классу. Оценить сбалансированность классов.
3. Если в наборе есть признаки, заведомо непригодные для использования в предсказательной модели, то удалить эти признаки из набора данных. В текстовой ячейке записать комментарий с обоснованием совершенного выбора.
4. Извлечь целевой признак *class* из набора данных в отдельную переменную, удалив его из набора данных.
5. Выполнить разовое разбиение набора данных и целевого признака на обучающую и тестовую выборки в соотношении 70/30, установив *random_state*, равный номеру своего варианта.
6. Подготовить признаки к использованию их в построении модели:
   - для входных признаков обосновать применение наиболее подходящего (с вашей точки зрения) метода масштабирования;
   - реализовать выбранные методы масштабирования;
   - для целевого признака обосновать, требуется ли для него кодирование категорий. Если да, то обосновать применение наиболее подходящего метода преобразования категорий в числовые значения (с вашей точки зрения) и реализовать его.

7. Создать модель *SGD*-классификатора, установив *random_state*, равный номеру своего варианта. Остальные параметры оставить со значениями по умолчанию. Обучить модель на обучающей выборке.
8. Оценить качество модели на тестовой выборке:
   - преобразовать тестовую выборку, применив обученные ранее масштабизаторы;
   - вывести матрицу ошибок;
   - используя *classification_report*, вывести значения основных метрик качества;
   - дать интерпретацию полученным оценкам.
9. Вывести список доступных параметров модели *SGDClassifier*.
10. Создать сетку параметров, включающую как минимум 4 вида функции потерь, два типа регуляризатора, не менее 10 значений коэффициента регуляризации от 10^{-6} до 10, и количество итераций без улучшения перед остановкой обучения от 5 до 10 с шагом 1.
11. Обосновать выбор метрики качества для подбора оптимальных значений гиперпараметров.
12. Создать объект *GridSearchCV* , передать ему созданный ранее классификатор и сетку параметров и обучить его на обучающей выборке, используя выбранную метрику. Предусмотреть вывод времени, затраченного на перебор по сетке (можно использовать *%%time*).
13. Получить оценки алгоритма по первым пяти, последним пяти и пяти лучшим наборам параметров.
14. Вывести лучший классификатор, лучший набор параметров и оценку лучшего классификатора в соответствии с заданной метрикой.
15. Вывести матрицу ошибок и значения основных метрик качества для лучшего классификатора на тестовой выборке. Сопоставить с результатами, полученными в п. 7.
16. Проанализировать полученные результаты, сделать выводы.
17. С помощью *RandomizedSearchCV* организовать случайный поиск по той же сетке. Вывести время, затраченное на случайный перебор.
18. Вывести те же показатели, что и в п. 12-13. Сопоставить полученные результаты и время на перебор с результатами, полученными при выполнении п. 12-13.
19. Сделать выводы, сформулировать рекомендации по использованию инструментов *GridSearchCV* и *RandomizedSearchCV*.