Лабораторная работа №3. Метод стохастического градиента для гребневой регрессии

Задание. Реализовать метод стохастического градиента для обучения многомерной линейной регрессионной модели с L2-регуляризацией (гребневая регрессия или ridge-регрессия).
Вход: Dataset (размеченная обучающая выборка с вещественными ответами), λ (параметр экспоненциального скользящего среднего), eps (параметр остановки)
Выход: R^2 (коэффициент детерминации); τ (коэффициент регуляризации), w_i (коэффициенты регрессионной модели)
Дополнительные условия.
1. Библиотеки: numpy, pandas, matplotlib, seaborn и т.д.
2. Dataset: выбрать из репозитория UCI с небольшим количеством количественных признаков. Исходный датасет разбивается на обучающую (*train*) и тестовую (*test*) выборку. Рекомендация: добавьте в датасет дополнительный фиктивный признак равный *-1* на всех объектах, соответствующий пороговому параметру (w_0) модели (для удобства реализации).
3. Функционал качества с регуляризацией (эмпирический риск) :

$$Q = \sum_{i=1}^{l} (\langle w, x_i \rangle - y_i)^2 + \frac{\tau}{2} \|w\|^2$$

4. Метод обучения: метод наименьших квадратов + стохастический градиентный спуск. Градиентный шаг для регрессии: $w_i = w_{i-1}(1 - h\tau) - (\langle w_{i-1}, x \rangle - y)x$
5. Тепм обучения: $h = 1/i$, где i – номер итерации (другие разумные варианты приветствуются).
6. Инициализация весов: $w_i = 0$ (другие разумные варианты приветствуются).
7. Порядок выбора объектов из обучающей выборки: случайный (другие разумные варианты приветствуются).
8. Критерий остановки: $Q_i - Q_{i-1} < eps$, где Q_i – сумма квадратов ошибки (невязки) на i-ой итерации, вычисляется (оценивается) по экспоненциальному скользящему среднему. Построить график сходимости Q_i.
9. Контроль переобучения: *L2*-регуляризация с параметром τ, который настраивается по контрольной выборке (в качестве контрольной выборки брать тестовую). Построить график зависимости качества модели на контрольной выборке от параметра τ. Диапазон и шаг сетки для оптимизации τ определить самостоятельно.
10. Показатель качества модели: коэффициент детерминации R^2 – доля объяснённой изменчивости ответов моделью, вычисляется как квадрат коэффициента корреляции между истинными (y) и предсказанными моделью значениями ответов ($\hat{y}$). Чем ближе к 1, тем выше качество модели.

Требования и рекомендации к реализации.

1. Рекомендуемые имена переменных: *X* – матрица объекты-признаки, *y* – ответы; *X_train, y_train* – объекты и ответы обучающей выборки; *X_test, y_test* – объекты и ответы тестовой выборки.
2. Модульная структура программы. Рекомендуемые функции: SGD (стохастический градиентный спуск), CrossValidation (скользящий контроль – оценка качества модели на контрольной выборке), Predict (получение ответа от модели). Можно реализовать в виде класса SGD_Ridge с соответствующими методами (ООП вариант приветствуется).
3. Применить возможности nampy, в.ч. метод «.dot» (скалярное произведение), сложение векторов («+») и умножение на число («*») и т.д. – намного упрощает реализацию.
4. Построить график зависимости между истинными и предсказанными ответами на контрольной выборке.
5. Описание датасета в начале программы.
6. Комментарии к коду.

Материалы.

3. Шпаргалки Python-DataScience: https://www.dropbox.com/sh/gmfsu39jqsagyq9/AADD2w4M3eUF2s1jn_Fk4AMXa?dl=0
4. Мануал по библиотекам Data science: https://scipy.org/
5. Сто заданий по Numpy (чем больше сделайте, тем лучше для вас): https://github.com/rougier/numpy-100
6. Лекция Воронцова К.В. «Курс Машинное обучение» 2019: https://www.youtube.com/watch?v=SZkrxWhI5qM&list=PLJOzdkh8T5krxc4HsHbB8g8f0hu7973fK – Машинное обучение. Линейные методы. К.В. Воронцов, Школа анализа данных, Яндекс.