



СПбГЭТУ «ЛЭТИ»
ПЕРВЫЙ ЭЛЕКТРОТЕХНИЧЕСКИЙ



А. Ю. Филатов

Задачи по методам обработки данных

Фрагмент конспекта

СПбГЭТУ «ЛЭТИ», 2022 г.





ЛАБОРАТОРНАЯ РАБОТА 1. МЕТОД НАИМЕНЬШИХ КВАДРАТОВ

1.1 Цель работы

Изучить метод наименьших квадратов и реализовать его.

1.2 Общие положения

Для выполнения работы вам необходимо знать метод наименьших квадратов.

В работе требуется написать программу на языке программирования Python.

При написании программы нельзя использовать встроенную в numpy функцию `numpy.linalg.lstsq`.

Проверка работы происходит через автоматизированную систему moodle.

1.3 Порядок выполнения

Дан набор точек, который получен зашумлением некоторой прямой с исходными коэффициентами.

Используя метод наименьших квадратов, вам необходимо реализовать функцию `solution`, которая будет возвращать исходные коэффициенты прямой.

1.4 Результат работы

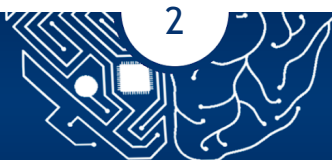
Пройдена проверка на moodle.

1.5 Контрольные вопросы и задачи

1. Методом наименьших квадратов посчитайте расстояние между функцией $y = x^2 + 1$ и набором точек.

Таблица 1

i	1	2	3	4	5	6	7	8	9	10
x_i	-0.5	0.81	0.99	0.6	5	2.9	1.4	3.2	0.01	2.13
y_i	-3.67	2.99	1.83	3.95	2.11	4.15	0.23	2.22	1.01	4.56





2. Назовите основное отличие интерполяции точек от их аппроксимации.
3. Для чего нужен resampling?

ЛАБОРАТОРНАЯ РАБОТА 2. ФИЛЬТР ЧАСТИЦ

2.1 Цель работы

Изучить алгоритм фильтра частиц и реализовать его.

2.2 Общие положения

Для выполнения работы вам необходимо знать алгоритм фильтра частиц.

В работе требуется написать программу на языке программирования Python.

Проверка работы происходит через автоматизированную систему moodle.

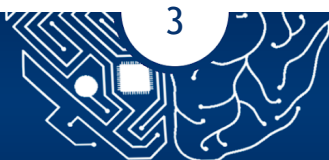
2.3 Порядок выполнения

Дана функция, $f(x) = c_0 e^{c_1 x + c_2} + c_3$ характеризующая поведение частицы, где вещественные числа c_0, c_1, c_3, c_4 - коэффициенты функции, записываются подряд в массив `theta`.

На вход программе подается целое число N - количество частиц и сгенерированный набор `thetas`. `thetas` содержит N массивов `theta`, описанных ранее. Требуется реализовать программу, которая будет производить фильтрацию частиц и выводить частицу с максимальным весом.

Для этого успешного решения задачи вам потребуется реализовать следующие функции:

1. `solution(N, thetas)` - главная функция программы, получает сгенерированные данные, выводит частицу с максимальным весом
2. `delta(t, y, theta)` - вычисляет “расстояние” от траектории частицы до исходной функции методом наименьших квадратов





3. `normalize_weights(weights)` - нормализует входной массив `weights` с весами частиц
4. `best_theta(thetas, weights)` - вычисляет индекс частицы с коэффициентами из `thetas`, которая имеет максимальный вес в `weights`, и выводит элемент массива из `thetas` по данному индексу.

Пример входных данных:

5, [[0.0313, 0.1137, 0.0070, 0.0029], [0.0022, 0.2, 0.0806, 0.0058], [0.135, 0.0017, 0.260, 0.0073], [0.0023, 0.0021, 0.0064, 0.0143], [0.0021, 0.1352, 0.1225, 0.0034]]

Пример выходных данных:

0.20965676125255983

2.4 Результат работы

Пройдена проверка на moodle.

2.5 Контрольные вопросы и задачи

1. Методом наименьших квадратов посчитайте расстояние между функцией $y = x + 2$ и набором точек из таблицы ниже.

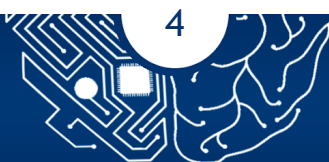
Таблица 2

i	1	2	3	4	5	6	7	8	9	10
x_i	-0.5	0.81	0.99	0.6	5	2.9	1.4	3.2	0.01	2.13
y_i	-3.67	2.99	1.83	3.95	2.11	4.15	0.23	2.22	1.01	4.56

2. Дан набор весов частиц. Для каждой частицы найдите вероятность, что она не отсеется при `resampling`'ге.

Таблица 3

i	1	2	3	4	5
w_i	0.14	0.25	0.19	0.33	0.09





3. Как связаны входное количество различных частиц и точность найденного результата?

ЛАБОРАТОРНАЯ РАБОТА 3. ФИЛЬТР КАЛМАНА

3.1 Цель работы

Изучить алгоритм фильтрации Калмана.

3.2 Общие положения

В работе требуется реализовать фильтр Калмана без матриц состояний на языке Python.

Проверка работы осуществляется на moodle.

3.3 Порядок выполнения

Используя теоретический материал курса, реализуйте фильтр Калмана с использованием матриц, отвечающих за:

- Нынешнее состояние (вектор входных наблюдений)
- Вектор входного воздействия
- Матрица связи X и предыдущего значения X
- Матрица, отвечающая за входное воздействие
- Матрица ковариации для X
- Ковариация шумов
- Часть модели, которая отвечает за шумы !!!
- Вектор наблюдений
- Связь между X и Z
- Ковариация шумов





Фильтр Калмана должен возвращать текущее состояние и степень доверия к нему.

3.4 Результат работы

Пройдена проверка на moodle.

3.5 Контрольные вопросы и задачи

1. Что такое степень доверия к предыдущему состоянию?
2. Чем отличается расширенный фильтр Калмана от обычного?
3. Привести пример задачи, которую можно решить при помощи фильтра Калмана?





ЛАБОРАТОРНАЯ РАБОТА 4. БАЙЕСОВСКИЙ ФИЛЬТР СПАМА

4.1 Цель работы

Изучить алгоритм Байесовского фильтра спама.

4.2 Общие положения

Для выполнения работы необходимо знать формулу условной вероятности по Байесу.

В работе требуется реализовать программу на языке Python.

Проверка работы осуществляется на moodle.

4.3 Порядок выполнения

Дана обучающая выборка, в которой каждое предложение может быть или спам, или не спам. Относительно данной выборки необходимо определить частоту появления каждого слова в спам предложениях и не спам предложениях. По полученным результатам необходимо применить формулу Байеса(1)

$$\Pr(S|W) = \frac{\Pr(W|S) \cdot \Pr(S)}{\Pr(W)} = \frac{\Pr(W|S) \cdot \Pr(S)}{\Pr(W|S) \cdot \Pr(S) + \Pr(W|H) \cdot \Pr(H)} \quad (1),$$

где

- $\Pr(S|W)$ - условная вероятность того, что сообщение спам, при условии, что слово W находится в тексте S ;
- $\Pr(S)$ - полная вероятность того, что произвольное сообщение спам;





- $\Pr(W|S)$ - условная вероятность того, что слово W появляется в сообщениях, если они являются спамом;
- $\Pr(H)$ - полная вероятность того, что произвольное сообщение не спам;
- $\Pr(W|H)$ - условная вероятность, что слово W появляется в сообщениях, если они не являются спамом.

4.4 Результат работы

Пройдена проверка на moodle.

4.5 Контрольные вопросы и задачи

1. Может ли $P(S)$ и $P(H)$ равны по 0.6?
2. Приведите пример апостериорных из формулы Байеса(1)
3. Продемонстрируйте на собственном примере как рассчитывается $\Pr(W|S)$.

