# OOPython

## Задача 7. Аппроксимация функций методом конечных элементов

### Элементы теории

Будем решать задачу аппроксимации заданной функции  кусочно-линейной функцией :

Изображение выглядит как карта, текст

Автоматически созданное описание

Точки  отрезка , в которых происходит излом функции , делят отрезок на подобласти – **конечные элементы.**

В пространстве кусочно-линейных функций  имеется базис из т.н. функций-«шляпок» . Т.о. искомую функцию  можно записать в виде суммы с неопределенными коэффициентами :



Задача состоит в нахождении этих коэффициентов, при которых функция  будет «близка» к заданной функции .

Изображение выглядит как объект

Автоматически созданное описание

### Методы нахождения коэффициентов

#### Метод наименьших квадратов

Будем искать коэффициенты исходя из формулы



Введем несколько обозначений:



Перепишем формулу с учетом новых обозначений:





Пропуская выкладки, для нахождения коэффициентов  получаем СЛАУ:



Решив СЛАУ и найдя коэффициенты , подставим их в формулу – получаем приближенное решение.

*Примечание:* в случае кусочно-линейной аппроксимации **матрица масс ** - симметричная трехдиагональная, значит, для повышения эффективности расчетов:

* вычислять требуется только элементы матрицы, расположенные на диагоналях
* для решения СЛАУ можно воспользоваться эффективным методом прогонки

#### L2-проекция

Будем искать коэффициенты исходя из формулы

,

что эквивалентно выполнению условий



Покажем это:

* если выполняется , то 
* в обратную сторону: если выполняется , то ;  - произвольное вещественное число.

Итак:





В матричном виде:



Получилась та же СЛАУ, что и в результате применения метода наименьших квадратов.

*Примечание:* скорее всего, исторически, сначала был получены формулы для элементов матрицы ** и вектора ** через метод наименьших квадратов. Исходя из этих формул, можно провести рассуждения в обратном порядке и получить исходную формулу метода – .

#### Интерполяция

Ищем коэффициенты исходя из равенств  и  в узлах интерполяции :



В матричной форме эта СЛАУ записывается в виде:



#### Регрессия

Ищем коэффициенты исходя из равенств  и  в узлах регрессии :



Эта СЛАУ – переопределенная, у нее нет решения в классическом понимании. Однако, c помощью метода наименьших квадратов для СЛАУ, можно найти ее псевдорешение – точку в , доставляющую минимум функции :



Приравниваем нулю производные по коэффициентам:



Пропуская выкладки, для нахождения коэффициентов  получаем СЛАУ:



### Задание

#### Определение классов

Для методов **наименьших квадратов**, **интерполяции** и **регрессии** реализовать соответствующие классы **FEMSolver***Name*, минимизировав суммарное число строк кода с помощью наследования. В качестве заготовки использовать класс **FEMSolverLeastSquares** из **lecture\_11\_approximation\_of\_functions.ipynb.**

#### Использование классов

Использовать созданные классы для нахождения функций-аппроксимаций функции . Число конечных элементов на отрезке: .

В одном графическом окне построить графики:

* функции-аппроксимации, полученной методом наименьших квадратов
* функции-аппроксимации, полученной методом интерполяции; точки интерполяции 
* функции-аппроксимации, полученной методом регрессии; точки регрессии 
* аппроксимируемой функции 