

Лабораторная работа №1

“Численные методы решения нелинейных уравнений”

Цель работы: изучение численных методов решения нелинейных уравнений
ознакомление с синтаксисом и базовыми операторами языка Python.

Ход работы:

Номер этапа	Описание проделанных действий	Результат этапа
1	Ознакомление с краткими теоретическими сведениями, с заданием. Просмотр конспекта в тетради. Чтение документации к библиотеке “seaborn”.	Составлен план выполнения работы, определены основные методы и инструменты. Изучены возможности "seaborn" для визуализации данных.
2	Выполнение первого и второго заданий лабораторной работы. Используется seaborn для визуализации графика исходной функции.	Построен график функции $f(x) = x^3 - 6x^2 + 5x$ на интервале $[-1, 6]$. На график нанесены границы интервалов, содержащих корни.
3	Написание алгоритмов численного поиска решений нелинейных уравнений. Опора на краткие теоретические сведения. Использование формул, представленных в руководстве к работе.	Реализованы три численных метода: бисекции, хорд и Ньютона. Каждый метод оформлен в виде отдельной функции с возможностью задавать количество итераций.
4	Ручное тестирование написанных алгоритмов, руководствуясь данными, полученными при вычислениях на бумаге.	Корректность работы алгоритмов подтверждена. Для тестового запуска с 50 итерациями получены значения, близкие к аналитическим корням (0, 1, 5).
5	Визуализация двух графиков, показывающих зависимость величины ошибки от количества итераций того или иного алгоритма	Построены сравнительные графики в линейной и логарифмической

	численного поиска решений нелинейных уравнений.	шкалах. Графики демонстрируют скорость сходимости каждого метода.
6	Анализ полученных данных, подведение итогов лабораторной работы.	Сформулированы выводы о скорости и надежности каждого метода.

Выводы:

- Метод Ньютона продемонстрировал наилучшую скорость сходимости — благодаря использованию производной он достигает высокой точности за меньшее число итераций. Однако он требует знания производной и может расходиться при неудачном выборе начального приближения.
- Метод хорд работает быстрее бисекции, но уступает ей в надёжности. В некоторых случаях (например, при неудачном поведении функции на интервале) метод может требовать больше итераций для достижения нужной точности.
- Метод бисекции продемонстрировал наибольшую надёжность среди рассматриваемых методов: при наличии смены знака функции на заданном интервале он гарантированно сходится к корню вне зависимости от особенностей поведения функции. Основным недостатком метода является низкая скорость сходимости — среди всех реализованных алгоритмов он показал наихудшую производительность.
- Все три метода успешно справляются с поставленной задачей и могут быть использованы для поиска корней нелинейных уравнений. Выбор конкретного метода зависит от требований к точности, скорости вычислений и наличия информации о производной функции.

Использованные в работе функции:

Функция	Назначение	Основные параметры
<code>pandas.DataFrame()</code>	Создание таблицы данных для визуализации	<code>data</code> — словарь или массив данных; <code>columns</code> — названия столбцов

seaborn.set_theme()	Установка общего стиля оформления графиков	style — название темы ("ticks", "whitegrid", "dark")
seaborn.set_palette()	Выбор цветовой палитры	Название палитры ("muted", "deep", "pastel", "viridis")
seaborn.lineplot()	Построение линейного графика	data — pandas.DataFrame; x, y — названия колонок; linewidth — толщина линии; color — цвет; label — подпись в легенде; ax — объект осей
seaborn.despine()	Удаление лишних границ у графика	top, right — True/False
matplotlib.pyplot.subplots()	Создание фигуры и осей для графика	figsize — размер фигуры в дюймах (ширина, высота)
matplotlib.pyplot.scatter()	Отображение точек на графике	x, y — координаты; color — цвет; s — размер точек; marker — форма маркера; linewidth — толщина контура; zorder — порядок отрисовки
matplotlib.pyplot.tight_layout()	Автоматическая настройка расположения элементов	без параметров
matplotlib.pyplot.show()	Отображение графика	без параметров
matplotlib.pyplot.legend()	Добавление и настройка легенды	loc — положение; title — заголовок легенды
set_title() (метод ax)	Установка заголовка графика	label — текст; fontsize — размер шрифта; fontweight — насыщенность; pad — отступ
set_xlabel() (метод ax)	Подпись оси X	xlabel — текст; fontsize — размер шрифта
set_ylabel() (метод ax)	Подпись оси Y	ylabel — текст; fontsize — размер шрифта
grid() (метод ax)	Включение и настройка сетки	visible — True/False; linestyle — стиль линии; alpha — прозрачность; color — цвет
tick_params() (метод ax)	Настройка параметров делений на осях	axis — ось; which — деления; labelsizе — размер подписей
set_yscale() (метод ax)	Установка типа шкалы по оси Y	value — тип шкалы ('linear', 'log', 'symlog')