Практическая работа 13.
Интерполяция и аппроксимация данных

Цель работы: изучение основных команд, используемых для интерполяции и аппроксимации данных в среде MatLab, приобретение навыков обработки данных с использованием функций интерполяции и аппроксимации в среде MatLab.

Теоретические сведения

Интерполяция и аппроксимация данных

Аппроксимация функций заключается в приближенной замене заданной функции $f(x)$ некоторой функцией $\varphi(x)$ так, чтобы отклонение функции $\varphi(x)$ от $f(x)$ в заданной области было наименьшим. Функция $\varphi(x)$ при этом называется *аппроксимирующей*. График аппроксимирующей функции может не проходить через узловые точки, но приближать их с некоторой среднеквадратической погрешностью. Типичной задачей аппроксимации функций является задача *интерполяции*. Необходимость *интерполяции* функций в основном связана с двумя причинами:

- функция $f(x)$ имеет сложное аналитическое описание, вызывающее определенные трудности при его использовании (например, $f(x)$ является спецфункцией: гамма-функцией, эллиптической функцией и др.).
- аналитическое описание функции $f(x)$ неизвестно, т.е. $f(x)$ задана таблично (например, для вычисления: значений $f(x)$ в произвольных точках, определения интегралов и производных от $f(x)$ и т. п.)

Интерполяция

Простейшая задача *интерполяции* заключается в следующем. Для заданных $n + 1$ точек $x_i = x_0, x_1, \ldots, x_n$, которые называются *узлами интерполяции,* и значений в этих точках некоторой функции $f(x_i) = y_0, y_1, \ldots, y_n$ построить полином $\varphi(x)$ (*интерполяционный полином*) степени n вида

$$\varphi(x) = a_n x^n + a_{n-1} x^{n-1} + \ldots + a_1 x + a_0, \qquad (1)$$

принимающий в узлах интерполяции x_i те же значения y_i, что и функция $f(x_i)$.

2.1 Полиномиальная аппроксимация данных измерений, которые сформированы в виде некоторого вектора *y*, при определенных значениях аргумента, образующих вектор *x* такой же длины, как и вектор *y*, осуществляется процедурой *polyfit(x, y, n)*, где *n* – порядок аппроксимирующего полинома. Результатом этой процедуры является вектор длиной *(n+1)* из коэффициентов аппроксимирующего полинома, содержащий коэффициенты полинома в порядке уменьшения степеней *x*. При *n=1* заданную зависимость можно аппроксимировать прямой, при *n=2* заданную зависимость можно аппроксимировать параболой, при *n=3* заданную зависимость можно аппроксимировать кубической параболой, при *n=4* заданную зависимость можно аппроксимировать параболой четвертой степени.

Для выполнения п. 1.1 необходимо задать интервал [*a*, *b*] для *x*, шаг изменения, определить вектор y_i и использовать процедуру *polyfit(x, y, n)* (например, *p1=polyfit(x, y, 1)* для *n=1*).

2.2 В пакете MatLab есть функция вычисления математического выражения при заданных значениях аргументов. Функция имеет вид: *polyval (p, x)*, где *p* – вычисляемая функция, *x* - вектор аргументов функции. Этой функцией можно воспользоваться для проверки достоверности результатов аппроксимации.

Для выполнения п. 1.2 необходимо задать интервал [*a*, *b*] для *x*, шаг изменения, определить вектор y_i, использовать процедуру *polyfit(x, y, n)*, задать интервал [*c*, *d*] для *x1*, использовать процедуру *polyval (p, x1)* (например, *p2=polyfit(x, y, 2)*; *y2=polyval (p2, x1)* для *n=2*). Построить графики заданной дискретной функции и полученных при аппроксимации полиномов (для *n=1*, *n=2*, *n=3*, *n=4*) на интервале [*c, d*]. Представлять графики разными цветами, типами точек и линий. Подписать оси. Подписать график. Сформировать легенду.

2.3 Сплайн-интерполяция используется для представления данных отрезками полиномов невысокой степени – чаще всего третьей степени. Свойства кубической сплайн-интерполяции:
- график кусочно-полиномиальной аппроксимирующей функции проходит точно через узловые точки;
- в узловых точках нет разрывов и резких перегибов функции;
- благодаря низкой степени полиномов погрешность между узловыми точками обычно достаточно мала;
- связь между числом узловых точек и степенью полинома отсутствует;
- появляется возможность аппроксимации функций с множеством пиков и впадин.

Реализуется сплайн-интерполяция следующей функцией: *spline(x, y, xi)*. Эта функция использует векторы *x, y*, содержащие аргументы функции и ее значения, и вектор *xi*, задающий новые точки. Для нахождения *yi* используется кубическая сплайн-интерполяция.

Для выполнения п. 1.3 необходимо задать интервал [*a, b*] для *x,* шаг изменения, определить вектор y_i, задать интервал [*c, d*] для *x1*, использовать функцию *spline(x, y, xi)* (например, *y1=spline(x, y, x1);*). Построить графики заданной дискретной функции и результаты сплайн-интерполяции на интервале [*c, d*]. Представлять графики разными цветами, типами точек и линий. Подписать оси. Подписать график. Сформировать легенду.

2.4 Для одномерной табличной интерполяции используется функция interp1 в следующем виде: *interp1(x, y, xi, method)*, где с помощью параметра *method* можно задать метод интерполяции:
- ' nearest ' – ступенчатая интерполяция;
- ' linear ' – линейная интерполяция (по умолчанию);
- ' spline ' – сплайн-интерполяция (кубическая);
- ' cubic ' – кубическая интерполяция (интерполяция многочленами Эрмита);

Для выполнения п. 1.4 необходимо задать интервал [*a, b*] для *x,* шаг изменения, определить вектор y_i, задать интервал [*c, d*] для *x1*, использовать функцию *interp1(x, y, xi, method)* (например, *y3= interp1(x, y, x1,* 'nearest'*);*). Построить графики заданной дискретной функции и результаты одномерной табличной интерполяции на интервале [*c, d*] (для всех методов). Представлять графики разными цветами, типами точек и линий. Подписать оси. Подписать график. Сформировать легенду.