

4.3 Обработка данных измерений

Допустим, что есть некоторая зависимость $y(x)$, заданная рядом точек

x	2	4	6	8	10
y	5.5	6.3	6.8	8	8.6

Ее можно задать в командном окне MatLAB как матрицу **xydata**, содержащую две строки – значения **x** и значения **y**:

```
>> xydata = [2 4 6 8 10; 5.5 6.3 6.8 8 8.6]
```

```
xydata =
```

```
2.0000 4.0000 6.0000 8.0000 10.0000  
5.5000 6.3000 6.8000 8.0000 8.6000
```

Средства для обработки данных:

size(xydata) – функция, предназначена для определения числа строк и столбцов матрицы **xydata**. Она формирует вектор **[n, p]**, содержащий эти величины:

```
>> size(xydata)
```

```
ans =
```


```
2 5
```

Либо:

```
>> [n, p] = size(xydata)
```

```
n = 2
```

```
p = 5
```



$max(V)$ – функция, где V – некоторый вектор, выдает значение максимального элемента этого вектора. Аналогично, функция **$min(V)$** извлекает минимальный элемент вектора V .

Функции **$mean(V)$** и **$std(V)$** определяют, соответственно, среднее значение и среднеквадратичное отклонение от него значений элементов вектора V .

$sort(V)$ – функция сортировки, формирует вектор, элементы которого расположены в порядке возрастания их значений.

$sum(V)$ – вычисляет сумму элементов вектора V .

$prod(V)$ – выдает произведение всех элементов вектора V .

$cumsum(V)$ – формирует вектор того же типа и размера, любой элемент которого является суммой всех предшествующих элементов вектора V (вектор кумулятивной суммы).

$cumprod(V)$ – создает вектор, элементы которого являются произведением всех предшествующих элементов вектора V .

diff(V) – функция численного дифференцирования, создает вектор, который имеет размер на единицу меньший размера вектора V , элементы которого являются разностью между соседними элементами вектора V .

Пример:

```
» v = [1 0.1 0.5 0.1 0.1 0.4];  
» disp(diff(v))  
-0.9000 0.4000 -0.4000 0 0.3000
```

trapz(x,y) – численное интегрирование методом трапеций – вычисление площади под графиком функции $y(x)$, в котором соседние точки, заданные векторами x и y , соединены отрезками прямых. Если первый вектор x не указан в обращении, по умолчанию допускается, что шаг интегрирования равняется единице.

Пример. Вычислим интеграл от функции $y = \sin(x)$ в диапазоне от 0 до π . (Его точное значение равно 2).

```
» x = 0:pi/100:pi;  
» y = sin(x);  
» disp(trapz(x,y))  
1. 9998
```

Замечания по разделу:

При использовании функций **size**, **max**, **min**, **mean**, **std**, **sort**, **sum**, **prod**, **cumsum**, **cumprod**, **diff** к матрицам нужно иметь в виду, что соответствующие описанные выше операции ведутся по отношению к каждому из столбцов заданной матрицы.

Пример:

```
» A = [1 3 4; 3 0 7; 4 1 7];
```

```
» max(A)
```

```
4 3 7
```

Если при обращении к функциям **max** и **min** указать второй выходной параметр, то он даст информацию о номерах строк, где находятся в соответствующие максимальные (или минимальные) значения.

Например:

```
>> [M, n] = max(A)
```

```
M = 4 3 7
```

```
n = 3 1 2
```

4.4. Аппроксимация и интерполяция данных

Полиномиальная аппроксимация данных измерений, которые сформированы как некоторый вектор Y , при некоторых значениях аргумента, которые образуют вектор X такой же длины, что и вектор Y , осуществляется процедурой

polyfit(X, Y, n). Здесь n - порядок аппроксимирующего полинома. Результатом действия этой процедуры является вектор длиной $(n + 1)$ из коэффициентов аппроксимирующего полинома.

Пример:

```
» x = [1 2 3 4 5 6 7 8];
```

```
» y = [-1.1 0.2 0.5 0.8 0.7 0.6 0.4 0.1];
```

```
» polyfit(x,y,1)
```

```
ans = 0.1143 - 0.2393
```

```
» polyfit(x,y,2)
```

```
ans = -0.1024 1.0357 -1.7750
```

```
» polyfit(x,y,3)
```

```
ans = 0.0177 -0.3410 1.9461 -2.6500
```

Это означает, что заданную зависимость можно аппроксимировать или прямой

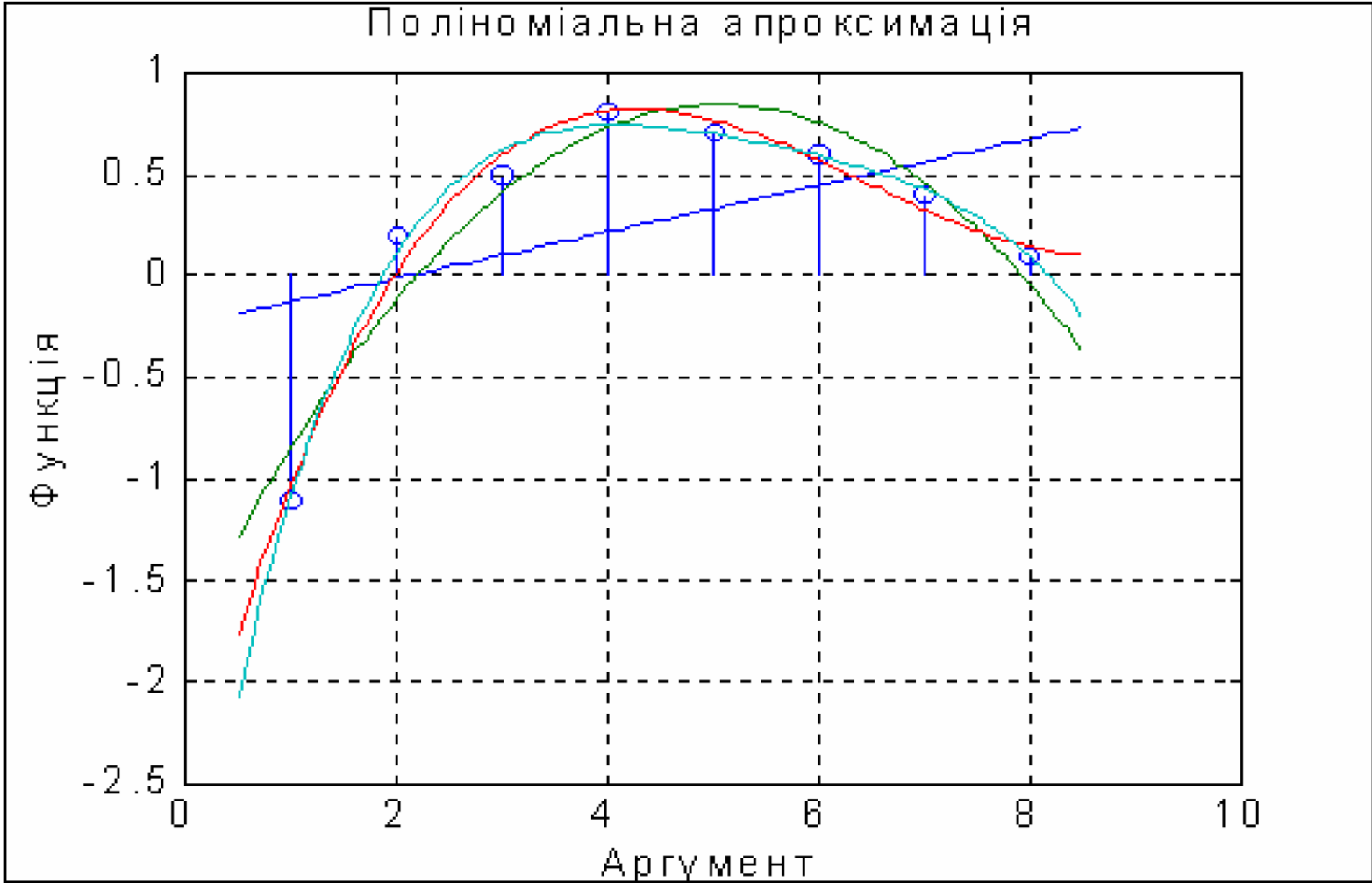
$$y(x) = 0,1143x - 0,2393,$$

или квадратной параболой

$$y(x) = -0,1024x^2 + 1,0357x - 1,775,$$

или кубической параболой

$$y(x) = 0,0177x^3 - 0,341x^2 + 1,9461x - 2,65$$





Одномерную табличную интерполяцию
осуществляет процедура *interp1*.

Обращение к ней в общем случае имеет вид:

$Y_i = \text{interp1}(X, Y, X_i, \text{'<метод>'}),$

и позволяет дополнительно указать метод интерполяции в четвертом входном аргументе:

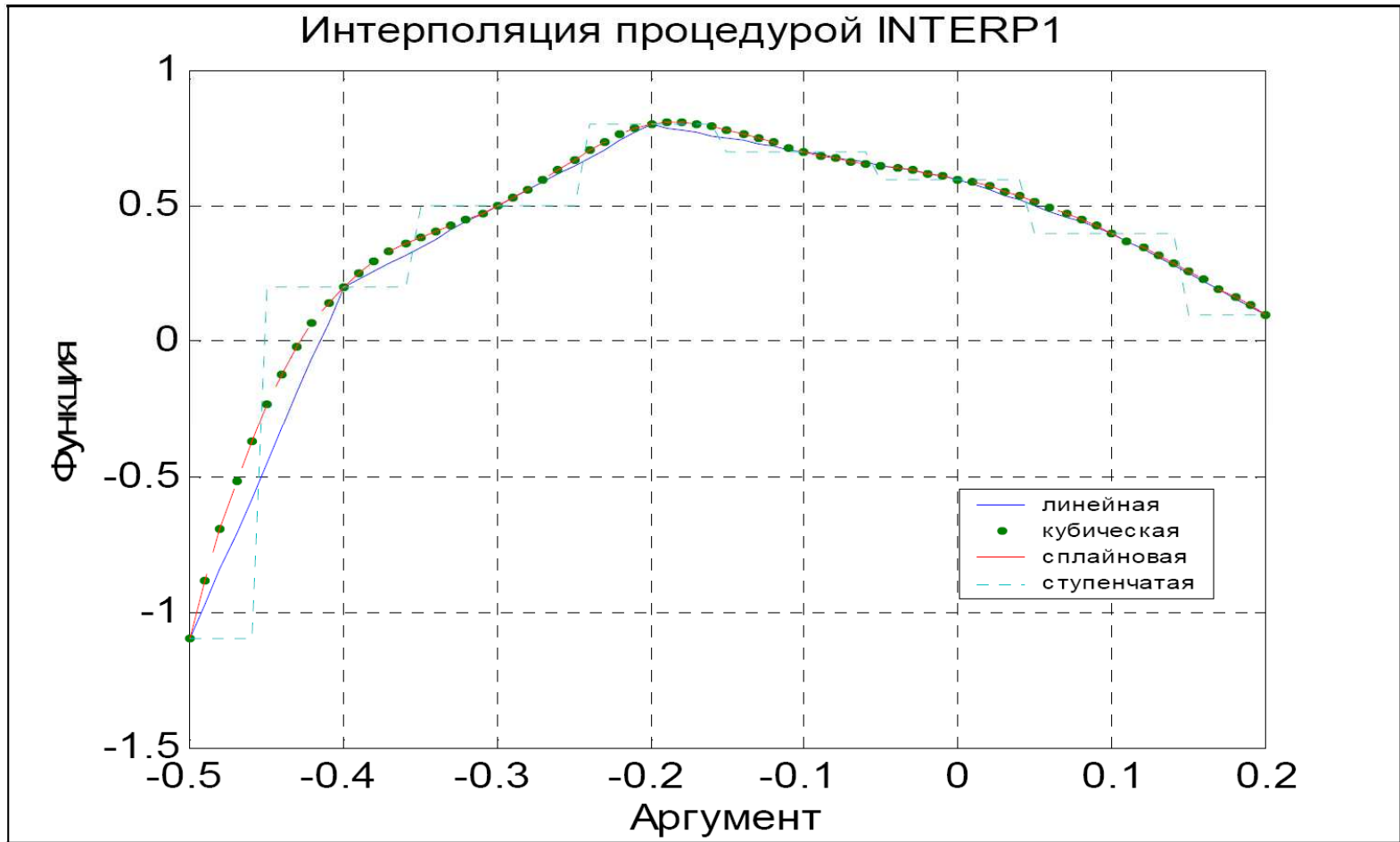
'nearest' - ступенчатая интерполяция;

'linear' - линейная;

'cubic' - кубическая;

'spline' - кубическими сплайнами.

Если метод не указан, осуществляется по умолчанию *линейная* интерполяция.



spline(X, Y, Xi) – осуществляет интерполяцию кубическими сплайнами. При обращении

$$Y_i = \text{spline}(X, Y, Xi)$$
