Если на некотором отрезке $[a \le x \le b]$ задан дискретный ряд значений y_i функции $y=f(x)$ в зависимости от значений аргумента x_i в виде числовой таблицы, то такая функция называется табличной. Тогда шагом таблицы является разница между соседними значениями аргумента $x_{i+1} - x_i = h$. И если эта разница постоянна, то говорят, что шаг таблицы равномерный, если нет, то шаг - неравномерный. При этом подразумевается, что аргументы таблицы являются точными числами.

Табличную функцию можно сформировать по результатам измерений или путем вычисления значений аналитически заданной функции. Аналитическим приближением табличной функции $f(x_i)$ является аналитически заданная и достаточно просто вычисляемая функция $p(x)$, совпадающая или близкая по своим значениям с табличными значениями функции $f(x_i)$ на всем отрезке области определения или на некоторой ее части

$$f(x_i) \approx p(x), \quad [a \le x_i \le b]$$

Процедура замены табличной функции $f(x_i)$ на аналитически заданную функцию $p(x)$, называется аппроксимацией, а сама функция $p(x)$ - аппроксимирующей.

Аппроксимация табличной функции называется интерполированием, или интерполяцией, если её значения y_i совпадают со значениями аппроксимирующей функции $p(x_i)$ в точках, соответствующих табличным аргументам x_i . В задачах интерполяции функцию $p(x)$ называют интерполирующей функцией, а табличные аргументы $x_0, x_i..., x_n$, в которых её значения должны совпадать со значениями табличной функции $f(x_i)$ - узлами интерполирования.

Способ интерполяции предполагает точное совпадение значений табличной и интерполирующей функций в узлах интерполирования. Если же табличная функция получена вычислениями с погрешностями или в результате выполнения измерений, то условие обязательного совпадения значений функций означало бы повторение имеющихся в расчетных или экспериментальных данных ошибок. В этих случаях, процедура аппроксимации должна решать задачу сглаживания табличных значений, правильно отражая описываемую зависимость. Способ сглаживающей аппроксимации, при котором график аппроксимирующей функции проходит по возможности близко между значениями табличной функции, называется фитированием, а сама аппроксимирующая функция – функцией фитирования.

Общая постановка задачи фитирования: для таблично заданной функции $f(x_i)$ требуется построить аналитическую функцию $\varphi(x,c_0,c_1,\ldots,c_m)$,так чтобы минимизировать значение функционала Q, сформированного совокупностью отклонений, имеющих место при фитировании значений табличной функции: min $Q\{\varphi(x,c_0,c_1,\ldots,c_m)-y_i\}$, где x_i , y_i - табличные значения, $c_0,c_1,\ldots,c_m$ - числовые коэффициенты.

Существует достаточно много способов интерполяции: *линейная*, *метод конечных разностей*, интерполяционная формула *Ньютона*, многочлен *Лагранжа, сплайнами*. Объединяет их то, что интерполирующую функцию ищут в виде многочлена (полинома). При этом интерполирование называют полиномиальным, а соответствующий многочлен – интерполяционным. Наиболее общей формулой для интерполирования является интерполяционный многочлен *Лагранжа*. Если узлы интерполирования $x_o, \ldots, x_n$ - равноотстоящие, то используются более простые для вычислений интерполяционные формулы *Ньютона*. В случае использования большого числа узлов интерполяции на всем отрезке [a, b], интерполяционные формулы *Лагранжа, Ньютона* могут привести к плохому приближению из-за накопления погрешностей в процессе вычислений. Кроме того, из-за расходимости процесса интерполяции увеличение числа узлов не обязательно приводит к повышению точности. Для снижения погрешностей при большом числе узлов интерполирования выполняется кусочно-полиномиальная интерполяция функции $f(x_i)$ с использованием сплайнов.

В случае, если стоит задача фитирования табличной функции, чаще всего используется метод наименьших квадратов.

В Scilab методы интерполяции реализованы с помощью большого набора встроенных функций. Перечислим некоторые из них:

interp - интерполирование функции, заданной дискретно, с помощью кубического сплайна

interp1 - одномерная функция

interpln - линейная интерполяция

linear_interpn - n размерная линейная интерполяция

smooth - сглаживание с помощью сплайнов

splin - кубическая сплайн интерполяция

Для задач фитирования функций можно использовать встроенную функцию **datafit()**, либо **- leastsq()**.

Пример 1. Даны значения табличной функции в точках (x,y): (3.4, 0.294118), (3.5, 0.285714), (3.6, 0.277778). Интерполировать данную функцию полиномом и найти её значение в точке x=3.55.

Представим нашу функцию в виде полинома: y = $P_2(x) = a_0 + a_1x + a_2x^2$, поскольку заданы только три точки. Чтобы найти коэффициенты полинома, необходимо решить систему из трёх уравнений:

$$\begin{cases} a_0 + a_1x_0 + a_2x_0^2 = y_0 \\ a_0 + a_1x_1 + a_2x_1^2 = y_1 \\ a_0 + a_1x_2 + a_2x_2^2 = y_2 \end{cases}$$

Решение выглядит следующим образом:

```
x = [3.4 3.5 3.6]; y = [0.294118, 0.285714, 0.277778];
[m1,n1]=size(x);
m = n1;
n = m - 1; //порядок полинома
B = ones(m,m); //приравнять коэфф. матрицы 1.0
//вычислить значения оставшихся коэфф. матрицы
for i = 1:m
for j = 2:m
B(i,j) = x(i)^(j-1);
end;
end;
a = linsolve(B,-y'); //решение системы
a=a'; //показать решение как вектор-строку
 a  =
   0.858314  - 0.2455    0.0234
 -->p = poly(a,'x','coeff')// полученный полином
 p  =
                                   2
   0.858314 - 0.2455x + 0.0234x
-->horner(p,x)
 ans  =
   0.294118    0.285714    0.277778 //значения в точках x
-->x1=3:0.1:5;
-->y1=horner(p,x1);
-->plot(x1,y1)
-->plot(x,y,'dk')
-->plot(3.55,horner(p,3.55),'or')
```

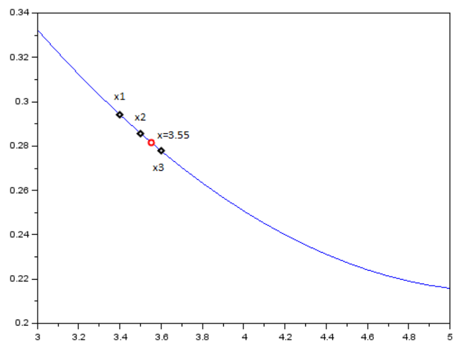

Рис.59 График интерполяционного полинома с узлами интерполяции

Пример 2. Написать файл-сценарий для интерполяции табличной функции полиномом Лагранжа порядка n. Вычислить значения табличной функции в точке x=3.44 в случае линейной и квадратичной интерполяции для значений табличной функции в точках (x,y): (3.4, 0.294118), (3.5, 0.285714), (3.6, 0.277778).

```
n=1;// степень полинома
x = [3.4 3.5 3.55 3.65]; yy = [0.294118, 0.285714, 0.281690, 0.273973];
x0=3.44
[m1,n1]=size(x);
m = n1;
if(n>m-1) then
error('используйте меньшую степень полинома n');
abort
end
m = n + 1;
N = ones(1,m);
D = N;
L = N;
y = 0.0;
for j = 1:m
for k = 1:m
if(k <> j) then
N(j) = N(j)*(x0-x(k));
D(j) = D(j)*(x(j)-x(k));
end;
end;
L(j) = N(j)/D(j);
y = y + L(j)*yy(j);
end;
 y  =
   0.2907564  //линейная n=1
  y  =
  0.2906994 // квадратичная n=2
```

Пример 3. Задана табличная функция в точках (x,y): (-1, 1), (1, 3), (2, -1),(3.5,1.9),(5,4). Линейно интерполировать функцию и построить её график на интервале [-4; 45], найти ее значение в точке x = 1.5.

```
x=[-1 1 2 3.5 5]; y=[-1 3 -1 1.9 4];
xy=[x;y];
x_new=(-4:8);
yi=interpln(xy,x_new);
plot2d(x',y',-2);
plot2d((-4:8)',yi');
-->z=interpln(xy,1.5)//значение в точке x=1.5
 z  =
    1.
```

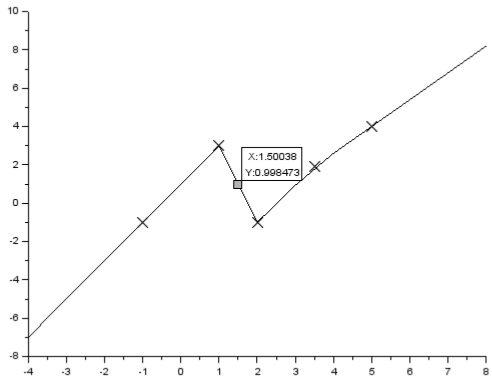

Рис.60 График табличной функции зад.3

Пример 4. Задана табличная функция (значения tg(x)) в точках (x,y): (-1.5, -14.1014), (-0.75,-0.931596), (0, 0), (0.75 ,0.931596),(1.5,14.1014). Интерполировать функцию линейно, кубическим сплайном, по соседним точкам и построить её график на интервале [-2; 2], найти ее значение в точке x = 1.5. Использовать функцию **interp1()**.

```
x=[-1.5 -0.75 0 0.75 1.5];
y=[-14.10114 -0.931596 0 0.931596 14.1014];
xx=linspace(-2,2,100);
yy1=interp1(x,y,xx,'linear');
yy2=interp1(x,y,xx,'spline');
yy3=interp1(x,y,xx,'nearest');
plot(xx',[yy1';yy2';yy3]',x,y,'*')
xtitle('interpolation of square function')
legend(['линейная','сплайн','соседняя'],a=2)
-->yy2=interp1(x,y,1.5,'spline')//значение в x=1.5
 yy2  =
    14.1014
```

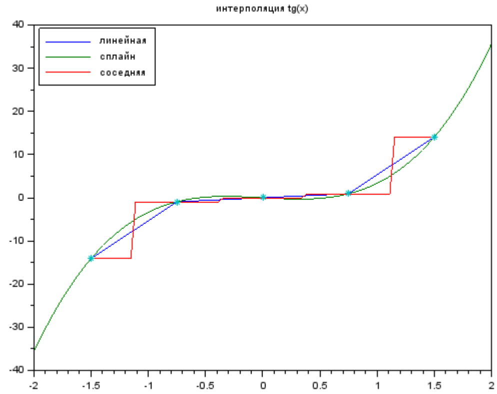

Рис.61 График y=tg(x)

Пример 5. Задана табличная функция в точках (x,y): (0.0, 15), (1.2, 29), (3.5, 13.3), (4.2 ,-6.41),(6.2,2.9),(8.1,17.1),(11.2,-8) Интерполировать её кубическими сплайнами и построить график, полученной функции.

Чтобы интерполировать функцию кубическими сплайнами в Scilab, необходимо вначале вычислить коэффициенты сплайна с помощью функции **d=splin(x,y)**, а затем вычислить значения интерполяционного полинома в нужной точке, используя функцию **y=interp(t,x,y,d)**.

```
y = [15.0,29.0,13.3,-6.4,2.9,17.1,-8.0];
x = [0.0,1.2,3.5,4.2,6.2,8.1,11.2];
d = splin(x,y);
xx = [0:0.1:11.2];
[y0,y1,y2,y3] = interp(xx,x,y,d);
plot(xx,y0)
plot(x,y,'ok')
```

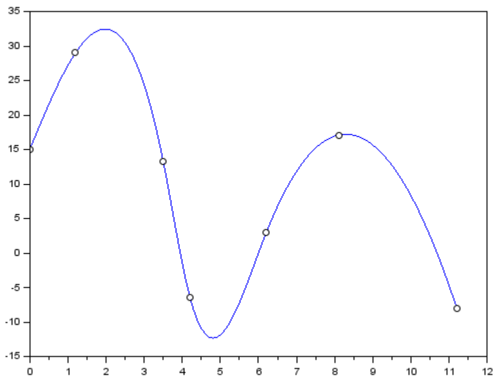

Рис.62 График функции

Пример 6. Задана табличная функция в точках (x,y): (0.0, 15.5), (1.25, 29.1), (3.75, 13), (4.32 ,-6.9),(6.2,2.99),(8.1,15.1),(11.25,-5) Интерполировать её кубическими сплайнами, вычислить значения функции с шагом Δx=0.1, и построить график, полученной функции.

Для решения этой задачи используем функцию Scilab **smooth()**.

```
y = [15.5,29.1,13.0,-6.9,2.99,15.1,-5.0];
x = [0.0,1.25,3.75,4.32,6.2,8.1,11.25];
xy_o=[x;y];
xy_f=smooth(xy_o,0.1);
xx=xy_f(1,:); yy = xy_f(2,:);
plot(xx,yy)
plot(x,y,'ok')
```

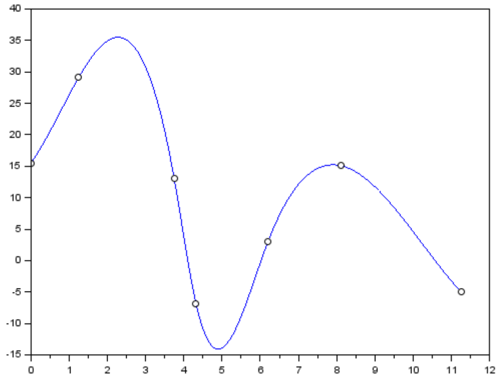

Рис.63 График функции

Пример 7. Задана функция $z = 2{\cdot}sin(x){\cdot}cos(y+x)$ где x и y меняются в пределах от 0 до 2π. Заданы узлы интерполирования- оси x и y равномерно разбиты на 10 отрезков. Используя линейную интерполяцию заданной функции, вычислить её значения в заданных точках (количество точек по оси x и y равно 50). Построить график интерполированной функции и отметить узлы интерполирования.

```
n = 10;
x = linspace(0,2*%pi,n); y = x;
z = 2*sin(x')*cos(y+x);
xx = linspace(0,2*%pi, 50);
[xp,yp] = ndgrid(xx,xx);
zp = linear_interpn(xp,yp, x, y, z);
clf()
plot3d(xx, xx, zp, flag=[2 6 4])
[xg,yg] = ndgrid(x,x);
param3d1(xg,yg, list(z,-5*ones(1,n)), flag=[0 0])
```

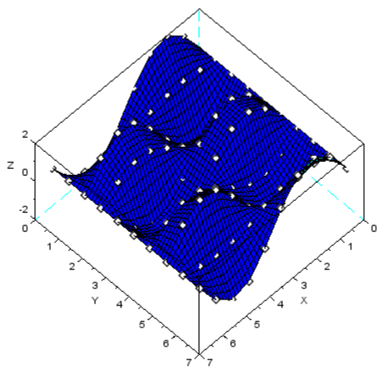

Рис.64 График функции $z = 2{\cdot}sin(x){\cdot}cos(y+x)$

Пример 8. Задан набор экспериментальных данных y, зависящих от x. Известно, что экспериментальные значения y_i содержат ошибки. Считая, что реальная зависимость (x_i,y_i) линейная $y_i = b+ax_i$, найти параметры a_1 и a_2. Значения x: 0; 0.15; 0.31; 0.48; 0.61; 0.77; 0.92; 1.01; 1.16; 1.32; 1.47. Значения y: 0; 0.16; 0.33; 0.42 ;0.59; 0.72 ;0.75; 0.88; 0.95; 0.96; 0.99.

Такая задача называется парной линейной регрессией или фитированием данных линейной зависимостью. Регрессионные коэффициенты a_1 и a_2 найдём с помощью функции Scilab [**a b sig**]**= reglin (x,y)**, где a,b –регрессионные коэффициенты, x и y – экспериментальные данные, sig-стандартное отклонение разности между данными и вычисленными точками.

```
x=[0 0.15 0.31 0.48 0.61 0.77 0.92 1.01 1.16 1.32 1.47];
y=[0 0.16 0.33 0.42 0.59 0.72 0.75 0.88 0.95 0.96 0.99];
plot2d(x,y,-5);
[a b sig]=reglin(x,y)
t=0:0.01:1.5;
yt=b+a*t;
plot2d(t,yt);
-->sig
 sig  =
    0.0690468
```

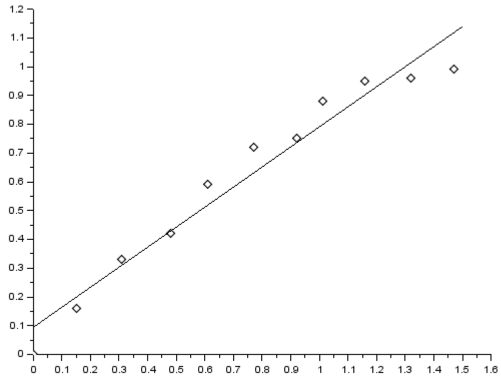

Рис.65 Линейная регрессия

Пример 9. Задан набор экспериментальных данных y, зависящих от t. Известно, что экспериментальные значения y_i содержат ошибки. Считая, что реальная зависимость (t_i,y_i) имеет вид $y = a\cdot exp(-b\cdot t)$, найти параметры a и b. Значения t: 0.25, 0.5, 0.75, 1.0, 1.25, 1.5, 1.75, 2.0, 2.25, 2.5. Значения y: 0.79, 0.59, 0.47, 0.36, 0.29, 0.23, 0.17, 0.15, 0.12, 0.08.

Данная задача отличается от предыдущей, тем, что зависимость y от t описывается нелинейной функцией. Поэтому, необходимо воспользоваться другой встроенной функцией - **leastsq()**. Эта функция минимизирует в задаче сумму квадратов разностей между экспериментальным и вычисленным значением (здесь wm – вес измерения).

$$f(x) = \sum_i wm^2(yth(tm(i),x) - ym(i))^2$$

```
function y=yth(t, x)
   y  = x(1)*exp(-x(2)*t)
endfunction
m = 10;
tm = [0.25, 0.5, 0.75, 1.0, 1.25, 1.5, 1.75, 2.0, 2.25, 2.5]';
ym = [0.79, 0.59, 0.47, 0.36, 0.29, 0.23, 0.17, 0.15, 0.12, 0.08]';
// веса измерений, здесь все равны 1
wm = ones(m,1);
//начальные значения параметров фита
x0 = [1.5 ; 0.8];
function e=myfun(x, tm, ym, wm)
   e = wm.*( yth(tm, x) - ym )
endfunction
[f,xopt, gopt] = leastsq(list(myfun,tm,ym,wm),x0)
tt = linspace(0,1.1*max(tm),100)';
yy = yth(tt, xopt);
plot(tm, ym, 'kx')
plot(tt, yy, 'b-')
legend(['заданные точки', 'фит']);
xtitle('фит функцией leastsq')
```

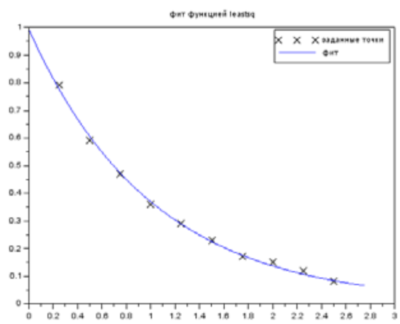

Рис.66 График фита функции с измеренными точками

Пример 10. Задан набор экспериментальных данных y, зависящих от x. Известно, что экспериментальные значения yi содержат ошибки. Считая, что реальная зависимость (x_i,y_i) имеет вид $y = Ax^2 + Bx + C$, найти параметры A, B и C. Значения x :72, 73,77, 81, 85, 92, 97. Значения y : 183, 172, 163, 153, 147, 141, 136.

Для решения этой задачи воспользуемся встроенной функцией - **datafit()**. Эта функция фитирует данные методом наименьших квадратов, минимизируя функцию ошибок e = G(p,z), где p – вектор –столбец из m рядов, представляющий параметры модели, а z – вектор-столбец из n рядов, представляющий переменные. Функция datafit ищет решение системы из k уравнений

$e_i = G(p,z_i) = 0$, минимизируя функционал

$$G^T(p,z_1)\cdot W\cdot G(p,z_1) + G^T(p,z_2)\cdot W\cdot G(p,z_2) + \ldots + G^T(p,z_n)\cdot W\cdot G(p,z_n),$$

где -W вес измерений. Простой вызов функции datafit выглядит следующим образом:

[p,err]=datafit(G,Z,p0)

Здесь G имя функции ошибок G(p,z), Z – матрица, ряды которой состоят из векторов переменных, p0 вектор-столбец, представляющий начальные значения параметров модели.

```
function [zr]=G(c,z)
zr=z(2)-c(1)-c(2)*z(1)-c(3)*z(1)^2
endfunction
x= [72 73 77 81 85 92  97];
y = [183 172 163  153 147 141 136];
z=[x;y];
c=[0;0;0];
[a,err]=datafit(G,z,c)
-->a
 a  =
    784.13675
  - 13.409302
    0.0695040
тогда  A = 0.0695040, B =  - 13.409302, C = 784.13675
dd2=0.0695040*x.^2-13.409302*x+784.13675;
   plot2d(x,dd2,-4);plot2d(x,dd2);
```

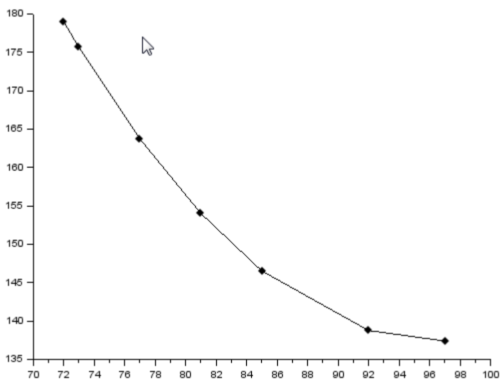

Рис.67 График фита функции с измеренными точками

Задачи.

1. Интерполировать таблично заданную функцию полиномами 1 степени.

x_1	x_2	x_3	y_1	y_2	y_3
-5	*-2*	*0*	*-2*	*0*	*-1*

2. Интерполировать таблично заданную функцию полиномом Лагранжа 2 степени.

x_1	x_2	x_3	y_1	y_2	y_3
-4	*-1*	*1*	*3*	*1*	*2*

3. Для функции $y=f(x)$, заданной значениями $f(1)=0.70$, $f(2)= 3.31$, $f(3)=0.181$ найти коэффициенты интерполирующего многочлена $P_2(x)=a_0+a_1x+a_2x^2$.

4. Дана таблица значений функции $y=lg(x)$;

x1	x2	x3	x4	x5	y1	y2	y3	y4	y5
11	12	13	14	15	1.0414	1.0792	1.1139	1.1461	1.1761

С помощью линейной интерполяционной формулой Лагранжа вычислить значение *lg(11.5)*, оценить погрешность вычисления.

5. Аппроксимировать таблично заданную функцию полиномами 1 степени.

x_1	x_2	x_3	y_1	y_2	y_3
-4	-3	-1	-2	0	-1

6. Интерполировать таблично заданную функцию с помощью команды **interpln**.

x_1	x_2	x_3	y_1	y_2	y_3
-3	-1	1	0	3	-2

7. Аппроксимировать таблично заданную функцию полиномом 2 степени.

x_1	x_2	x_3	y_1	y_2	y_3
-3	-1	0	1	-1	2

8. Восстановить многочлен по его значениям $y=P_3(x)$:

x_1	x_2	x_3	x_4	y_1	y_2	y_3	y_4
-1	0	1	2	1	1	-1	7

9. Найти многочлен P(x), данные о котором представлены следующей таблицей:

x	P(x)	P'(x)	P''(x)
-1	15	-14	-2
0	4	-7	
2	18		

10. Интерполировать таблично заданную функцию функцией **interpln().**

x_1	x_2	x_3	y_1	y_2	y_3
-2	*0*	*1*	*-1*	*1*	*-2*

11. Задана табличная функция

x_1	x_2	x_3	y_1	y_2	y_3
-5	*-3*	*-1*	*0*	*3*	*-2*

Интерполировать функцию линейно, кубическим сплайном, по соседним точкам и построить её график на интервале [-6; 0], найти ее значение в точке x = -1.5. Использовать функцию **interp1()**.

12. Задана табличная функция в точках (x,y):

x_1	x_2	x_3	x_4	y_1	y_2	y_3	y_4
-1	*0*	*1*	*2*	*1*	*1*	*-1*	*7*

Интерполировать её кубическими сплайнами и построить график, полученной функции.

13. Задана функция $z = 4 \cdot sin^2(x+y) \cdot cos(y+x)$ где x и y меняются в пределах от 0 до 2π. Заданы узлы интерполирования- оси x и y равномерно разбиты на 10 отрезков. Используя линейную интерполяцию заданной функции, вычислить её значения в заданных точках (количество точек по оси x и y равно 50). Построить график интерполированной функции и отметить узлы интерполирования.

14. Результаты измерения величин x и y даны в таблице:

x_i	-2	0	1	2	4
y_i	0.5	1	1.5	2	3

Предполагая, что между *x* и *y* существует линейная зависимость $y=a+bx$, определить коэффициенты *a* и *b*, использовать функцию [**a b sig]= reglin (x,y)**.

15. Задан набор экспериментальных данных y, зависящих от x. Известно, что экспериментальные значения y_i содержат ошибки. Считая, что реальная зависимость (x_i,y_i) имеет вид $y = C_1 x\cos(C_2 x) + C_3$;, найти параметры C_1, C_2, C_3.

x = 0 0.5 1 1.5 2 2.5 3 3.5 4 4.5 5 5.5 6

y= 5.02 6.08 3.33 -0.93 -0.22 7.83 16.52 15.55 2.67 -11.42 -11.78 5.09 25.25

Использовать функцию **leastsq()**.

16. Для табличной функции

x_i	0.5	1	1.5	2	2.5
y_i	10.5	1.6	0.55	0.26	0.15

подобрать подходящий вид аппроксимирующей её нелинейной зависимости из ряда:

а) $y=ax^b$ б) $y=ae^{bx}$ в) $y=a+b/x$ г) $y=1/(a+bx)$

Использовать при этом метод наименьших квадратов (функцию **datafit()**). Сравнивать между собой среднеквадратические погрешности.

17. Дана табличная функция

x	0	0.1	0.2	0.3	0.4	0.5	0.6	0.8	0.9	1.0
y	0	0.095	0.182	0.262	0.337	0.406	0.470	0.588	0.642	0.693

Используя функцию **reglin ()**, аппроксимировать данную функцию функцией вида $y=a_0+a_1x$ по трём точкам (x=0, x=0.5, x=1.0) и по всем точкам. Сравнить результаты с результатом применения функции **datafit()**. Построить график функции $y=ln(1+x)$ и график аппроксимирующей функции.

18. Заданы результаты измерений:

x = 1.30; 1.40; 1.55; 1.65; 1.74; 1.80; 1.95; 2.05; 2.13; 2.19; 2.35; 2.39; 2.58

y = 3.35; 3.45; 3.75; 4.30; 4.53; 4.86; 5.44; 6.04; 6.55; 7.32; 9.24; 10.21; 13.54

Считая, что реальная зависимость (x_i,y_i) имеет вид : $y = a_1 + a_2 x + a_3 x^2 + a_4 x^3$, найти значения параметров данной зависимости и построить графики экспериментальных данных и функции фитирования.

19. Построить уравнение аппроксимирующей параболы для следующих данных

x: 0, 1, 2, 3; y: -1.2, -0.1, 1.3, 1.8.

20. Найти параметры следующей функции $y=x/(Ax+B)$, если заданы её значения в 10 точках:

x	3	3.1	3.2	3.3	3.4	3.5	3.6	3.7	3.8	3.9
y	0.61	0.6	0.592	0.58	0.585	0.583	0.582	0.57	0.572	0.571