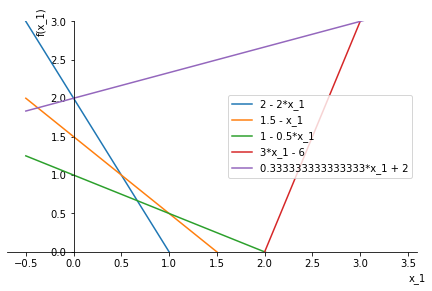
# Mathematical Methods in Engineering and Applied Science Problem Set 7

Kovalev Vyacheslav.

1. Solve the following LP problem both in Matlab using linprog (or elsewhere using an

analog of linprog) and directly by plotting the required regions:

minimize subject to:

After expression as a plot the lines:

Our solution lays in the largest inner triangle.

Min function - line with between itsalfe and

The solution is , that line the same as yellow.

Programming solution through the scipy.optimize.linprog :

fun: 1.5000000000042775, x=[0.5, 1]

1. Consider the data:

with the cubic, , chosen to fit them.

1. Find the best fit by using **fminsearch** starting with the initial condition (1,1,1,60).  
   I tried to minimise the error:

Error value = 6.123091

Iterations: 451  
A = 9.18552584e-03, B = -1.59522492e-01, C = -1.74893782e+00, D = 8.02351658e+01

1. Set up the normal equation and find the least-squares fit by solving it.  
   , where , , ,

A = 9.18550049e-03, B = -1.59521638e-01, C = -1.74894429e+00, D = 8.02351779e+01

Error = 6.123091284927192

1. Find the fit by using the genetic algorithm starting with the same initial condition

as in (a). If there is no convergence, or it is too slow, try other initial conditions.

Iterations: 500

Initial: as in (a)

Error = 22.14773768147146

A = 2.22958123e-02 B = -7.02423277e-01 C= 4.75604243e+00 D = 5.98320401e+01

1. Compare the computation times by the genetic algorithm with the fminsearch

method, for example using tic and toc commands in Matlab. The minima in both

cases have to agree with each other within 1%. You should use the same initial

conditions in both methods.

Best rersult from genetic alg:

Error = 6.33551427690039

A = 8.30041793e-03 B = -1.21700964e-01 C= -2.22105010e+00 D = 8.17972981e+01

Relative error = 3.46 %

Time = 8.9379 sec

Initial are as in (a)

relative error > than 1 % but I tried a lot with a lot of iterations, it is the best fit that I’ve gove.

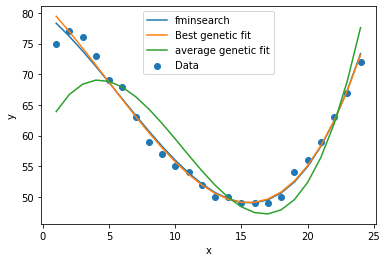
For fminsearch:

Time = 0.02535 sec

Tnitial are as in (a)

Error = 6.123091

Relative error

A= 9.18552584e-03 B = -1.59522492e-01 C = -1.74893782e+00 D = 8.02351658e+01 

1. Find the Fourier cosine series of and sine series of on . Which one does a

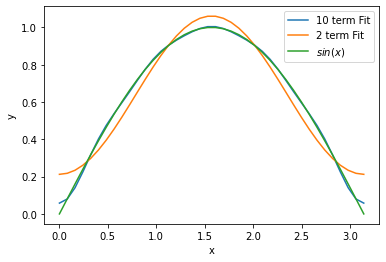
better job of representing its function and why? Make plots of 2-term as well as 10-term

approximations together with the original functions.

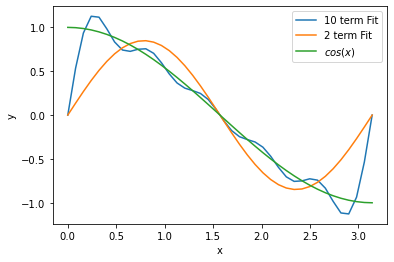
where

For cosine series of :

As result



For sine series of :



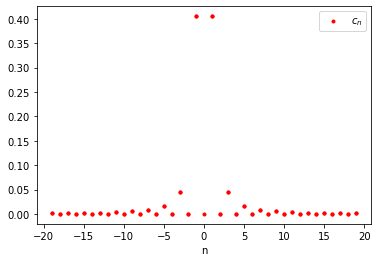
Sine series of cosine looks worse, because , while

1. Find the complex Fourier series of on . Plot the magnitude of the coefficients, , as a function of . What is ?

Let’s find magnitude of the coefficients,

Problem in

Other



– is the sum of all “energies”, I think because, analogically to vectors: square of vector is the sum of squares of orthogonal components of this vector:

, and I think converge to.

1. A harmonic oscillator is hit with the force , , . The equation of motion is and the initial conditions are .
2. Expand in the Fourier series in , plug it into the equation and determine the

solution. Assume that is not an integer. Plot the solution over

at including sufficient number of terms in the series.

multyply by and integrate over by .

Where

For we have:

Past this in our equation.

Now consider right part:  
 else if

else if

Let’s connect left and right parts:

If

Suppose than we have

If (p = 2n) then

Suppose is any

If (p = 2n+1) then

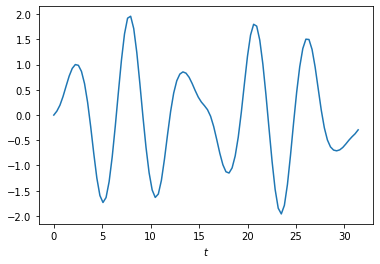
If(

Finally:

cosine is droped out because: if , then , so sum is 0.

Initial x(0) =0 =>

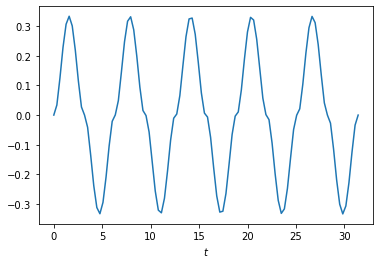
Initial

Plot:

1. (Optional, for extra credit). Redo part (a) for .

Consider

For for



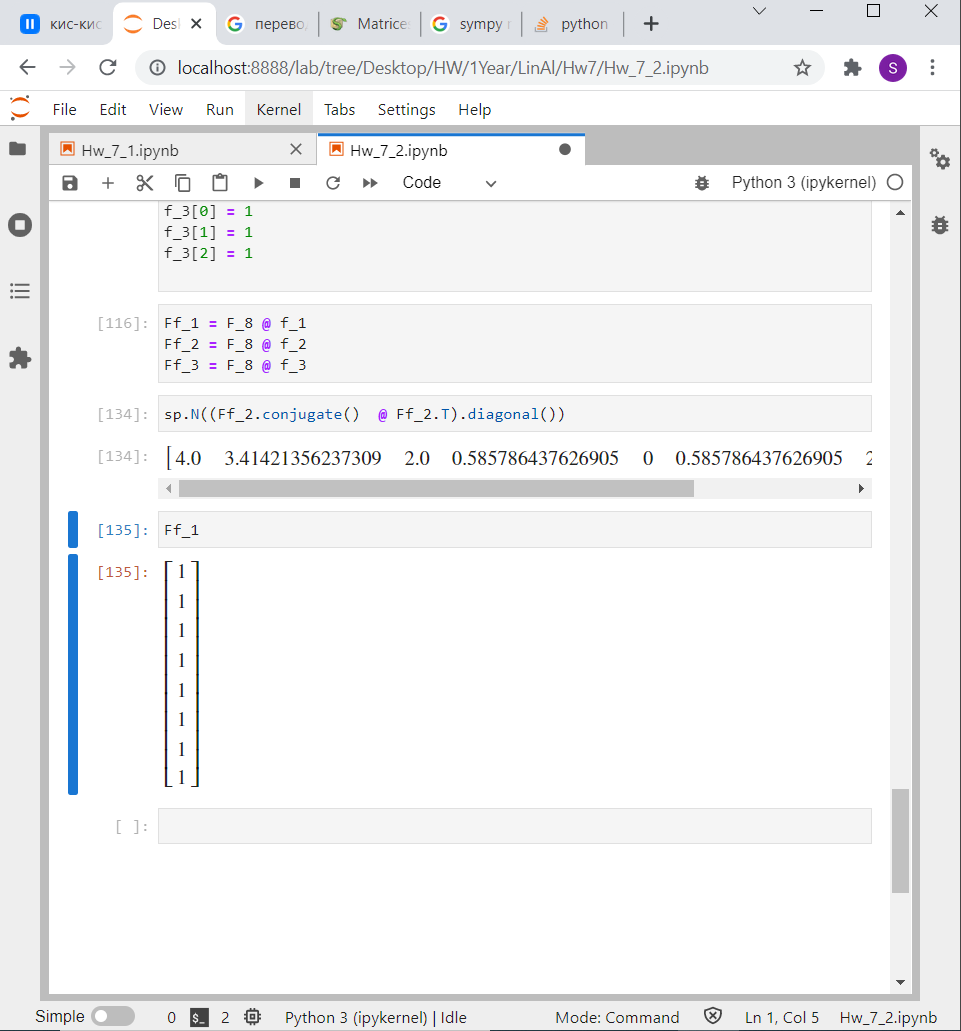
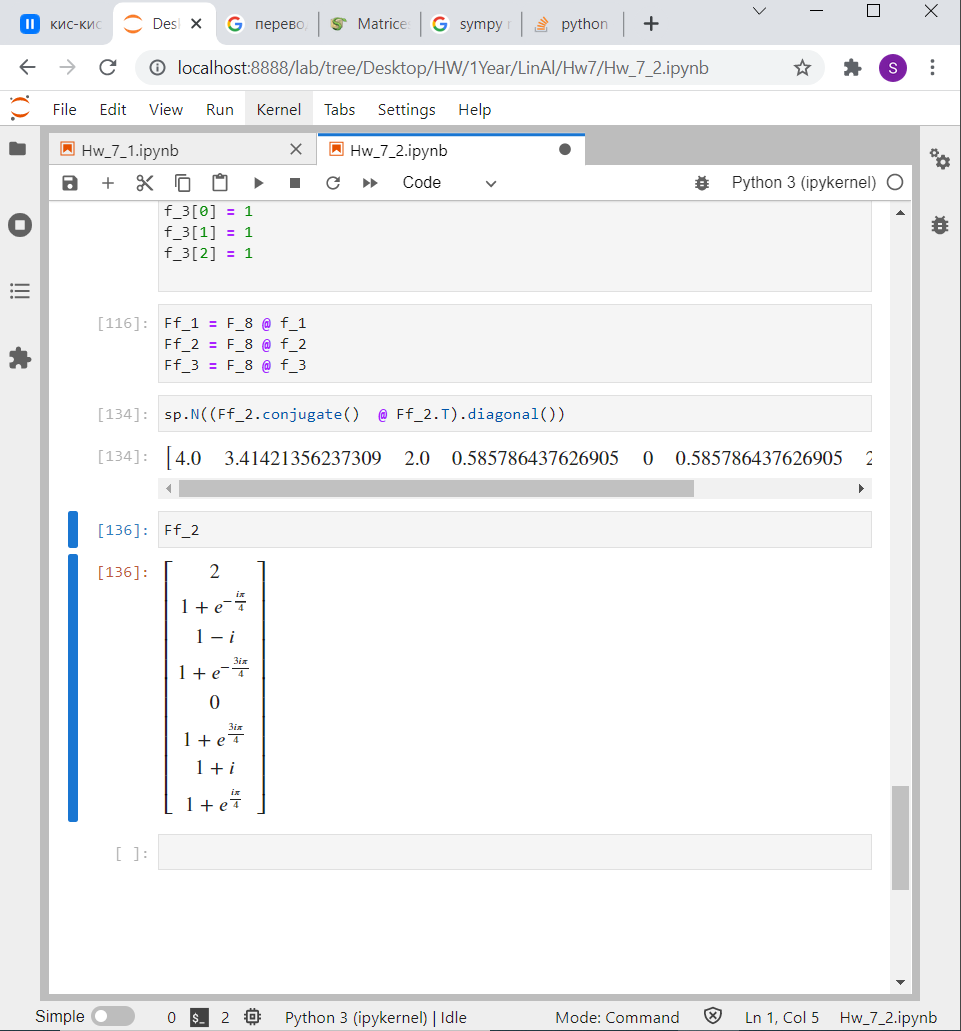
1. Write down the Fourier matrix F8 and decompose it into three factors containing F4.

Let f = zeros(8, 1) be a zero vector.

Where

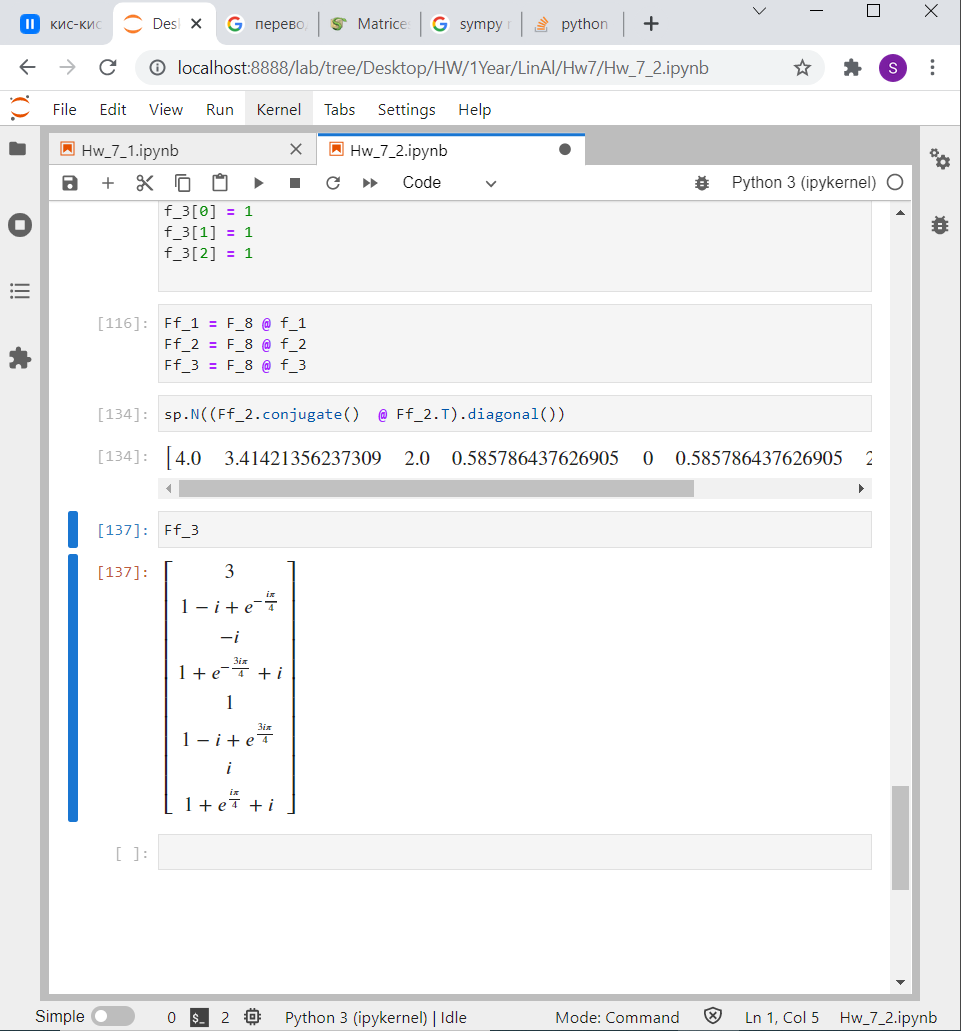
= even-odd permutation

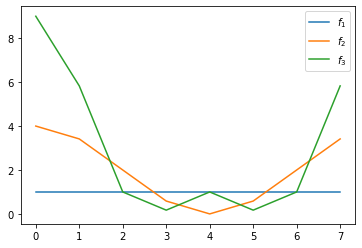
1. Modify f so that f (1) = 1, find the Fourier transform of and plot .   
   (b) Now let leaving the other components 0, and plot .   
   (c) Do the same with f (1 : 3) = 1. Explain your observations.



=

.





uses only one

uses

uses

1. (Optional, for extra credit). Given the data: x = 1 : 10, y = [0.78, 1.27, 1.33, 1.69, 1.96, 1.67, 2.07, 2.11, 1.91, 1.92], determine the least squares fit of the form y = a (1 − exp (−bx)) by setting up a nonlinear system of equations for a and b and solving it with Newton’s method (that you should implement yourself)

The most unstable method I’ve ever used. Through the attemts I came to algorithm:

For i in (1:20)

For j in (1:100)

Calculate

Choose such that F is min

For j in (1:100)

Calculate

Choose such that F is min

After that I obtained such that F is min.

Initial gues:

I tried to calculate and together but there was chaos.

Best Fit:   
F = 0.155785135254788 a = 2.00938295431986 b = 0.460632487826948

