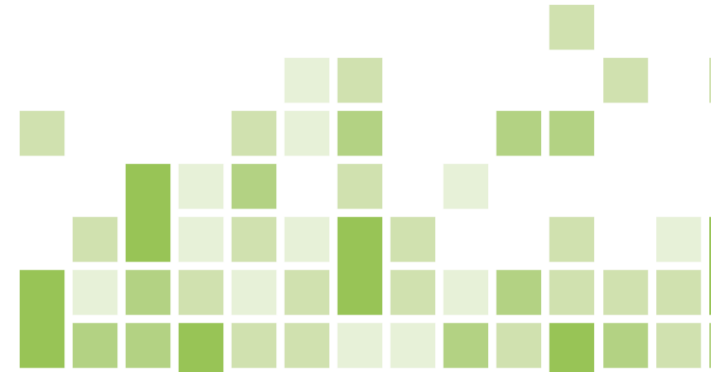




ТОМСКИЙ  
ПОЛИТЕХНИЧЕСКИЙ  
УНИВЕРСИТЕТ



МАТЕМАТИЧЕСКОЕ МОДЕЛИРОВАНИЕ ФИЗИЧЕСКИХ  
ПРОЦЕССОВ  
ЛЕКЦИЯ №3  
«Визуализация вычислений в Matlab»

Отделение ядерно-топливного цикла

Лектор:  
Зав. каф. - руководитель ОЯТЦ ИЯТШ  
Горюнов А.Г.

2020

# План лекции

3.1 Основы графической визуализации вычислений.

3.2 Построение двумерных графиков.

3.3 Построение трехмерных графиков.

3.4 Форматирование графиков.

3.5 Основы GUI (<https://www.youtube.com/watch?v=1KKAlyY3onI> ).

Информация по курсу:

<https://portal.tpu.ru/SHARED/a/ALEX1479/study/Matmod/Tab>

# 3.1 Основы графической визуализации вычислений

**MatLab** позволяет получать результаты в графическом виде при минимальных трудозатратах.

Основную работу, связанную с построением графика, масштабированием осей, подбором цветов, а в случае 3D графики и с углом обзора наблюдаемого объекта берет на себя среда MatLab.

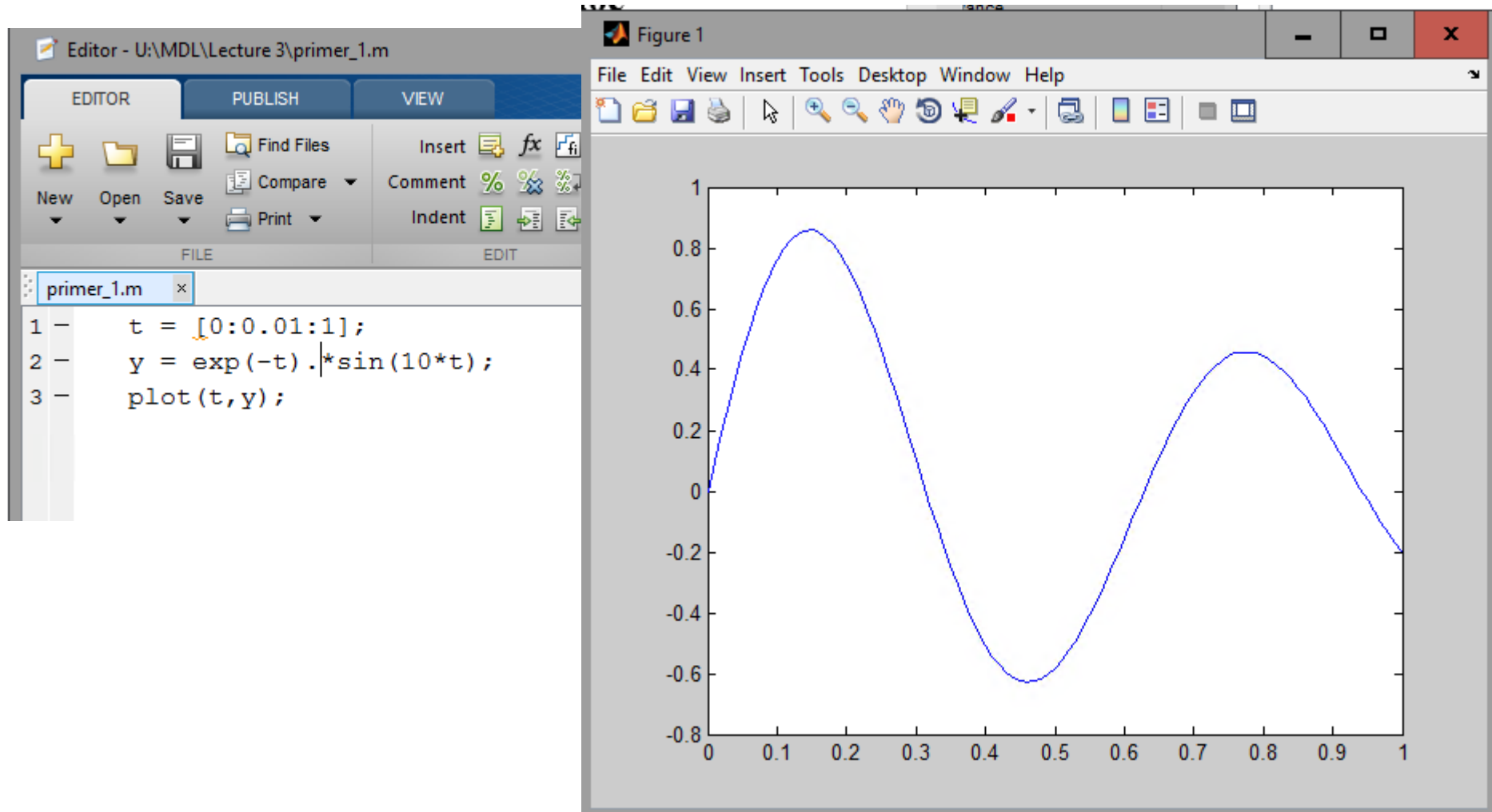
При создании с помощью компьютера графика той или иной зависимости необходимо иметь в виду, что компьютер способен работать **только с числами**, но не с непрерывными значениями аргументов и их функций.

Часто мы можем наблюдать, что график функции представлен на экране **плавными линиями** – это иллюзия. На самом деле между отдельными значениями аргументов и их функций выбран достаточно малый интервал, а разрешение монитора достаточно высокое. Если этот интервал увеличить, то на графике функции будут видны изломы.

# 3.2 Построение двумерных графиков

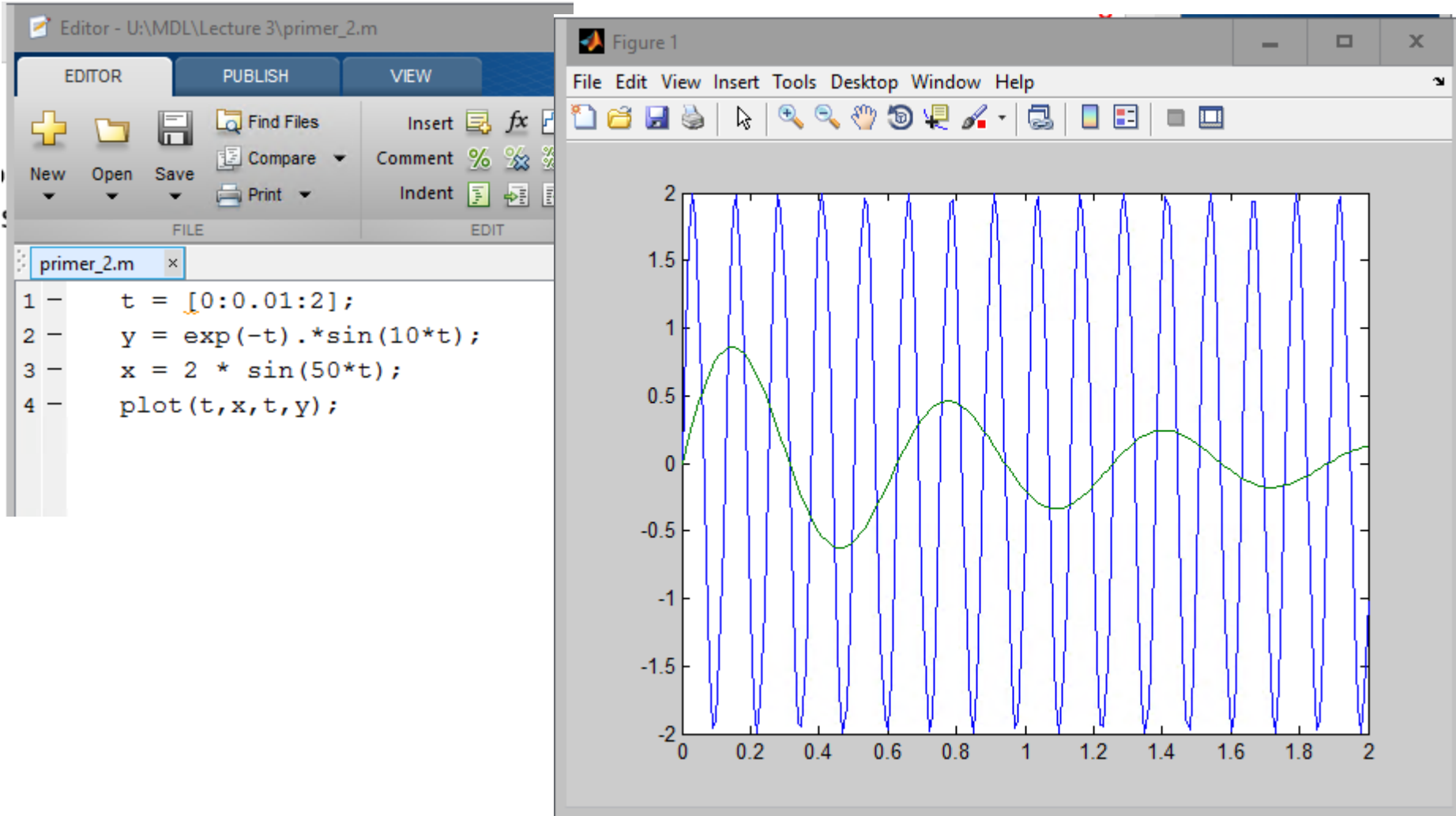
## 3.2.1 Графики в линейном масштабе

Построим график функции  $y(t) = e^{-t} \cdot \sin(10 \cdot t)$ , где аргумент  $t$  меняется в интервале  $[0,1]$  с шагом 0,01.



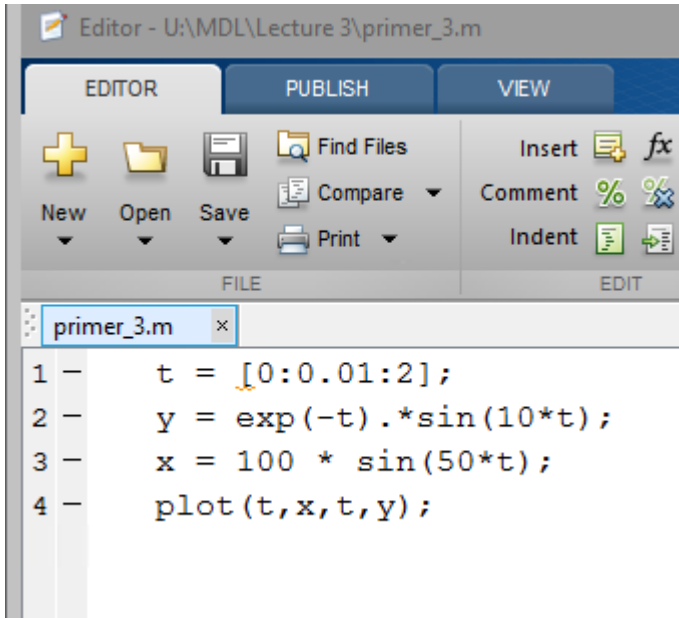
## 3.2.1 Графики в линейном масштабе

Построим в одном окне графики двух функций  $x(t) = 2 \cdot \sin(50 \cdot t)$  и  $y(t) = e^{-t} \cdot \sin(10 \cdot t)$ , где аргумент  $t$  меняется в интервале  $[0,2]$  с шагом  $0,01$ .

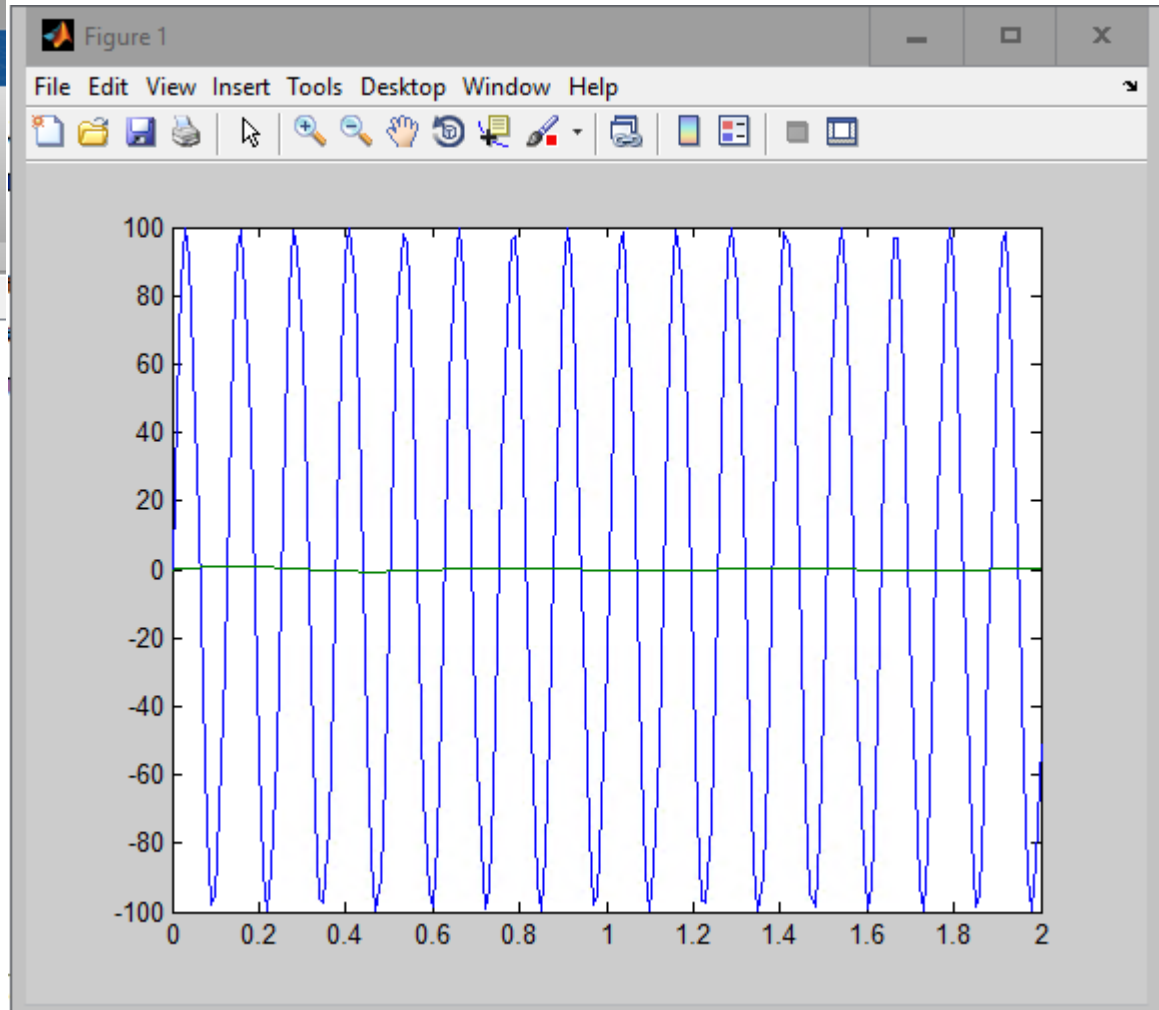


## 3.2.1 Графики в линейном масштабе

Попробуем в одном окне построить графики двух функций  $x(t) = 100 \cdot \sin(50 \cdot t)$  и  $y(t) = e^{-t} \cdot \sin(10 \cdot t)$ , где аргумент  $t$  меняется в интервале  $[0,2]$  с шагом 0,01.

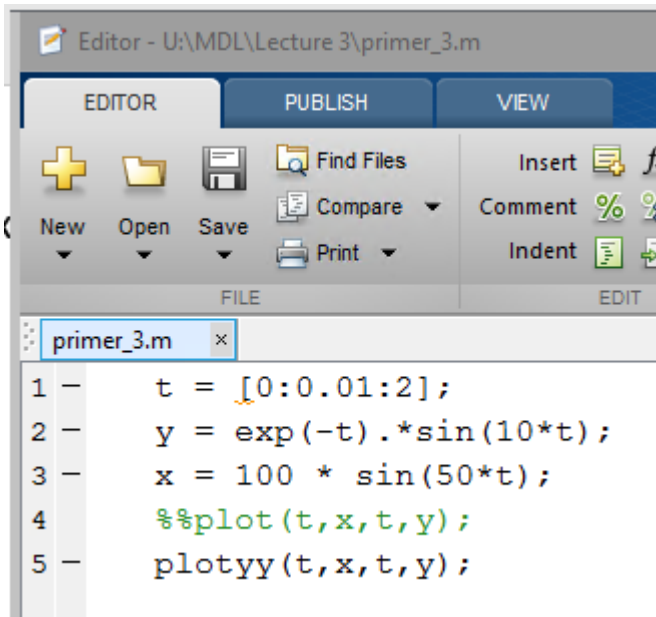


```
Editor - U:\MDL\Lecture 3\primer_3.m
EDITOR PUBLISH VIEW
+ New + Open + Save Find Files Compare Print
Insert Comment Indent
primer_3.m x
1 - t = [0:0.01:2];
2 - y = exp(-t).*sin(10*t);
3 - x = 100 * sin(50*t);
4 - plot(t,x,t,y);
```

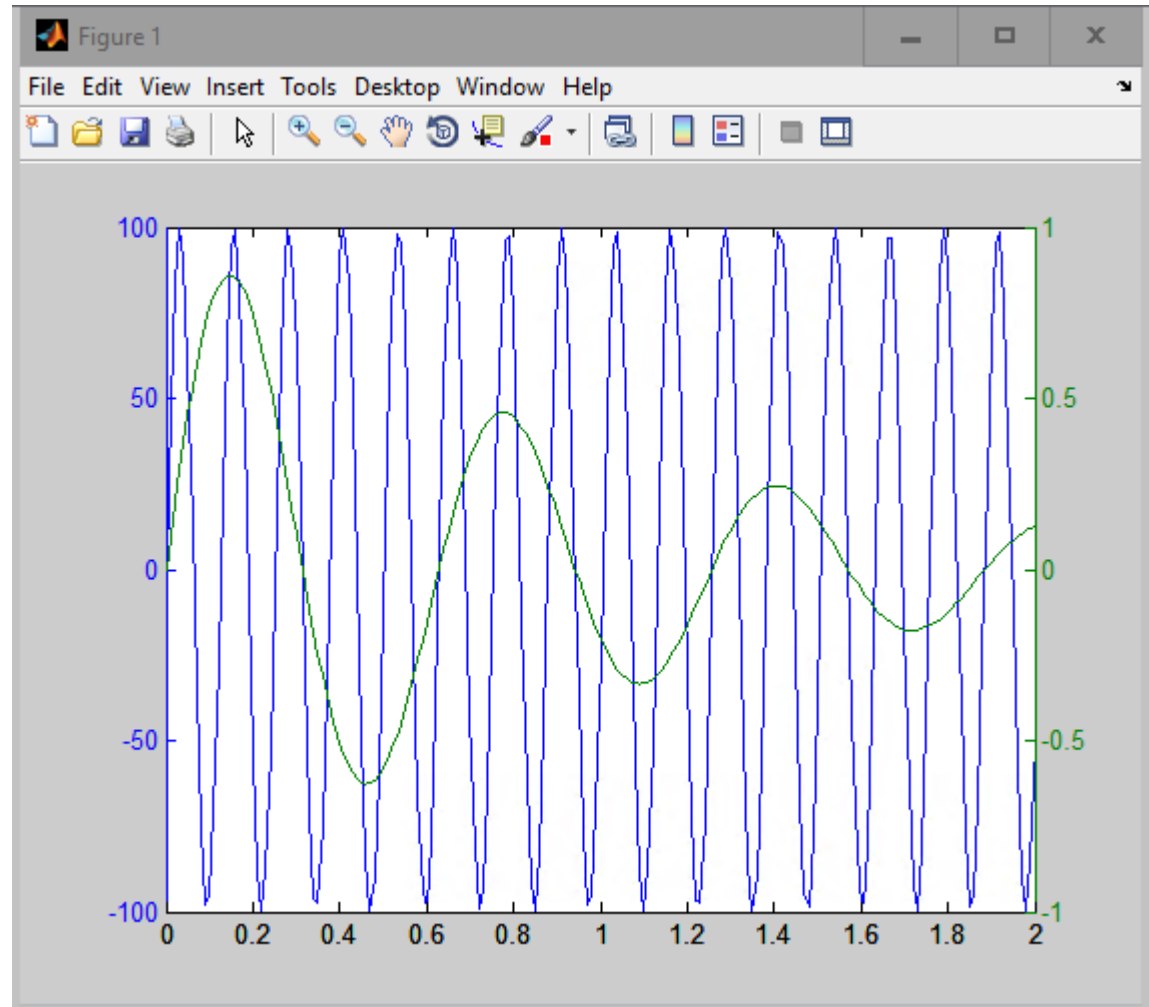


## 3.2.1 Графики в линейном масштабе

В таких случаях (см. слайд 6) лучше использовать функцию **plotyy**

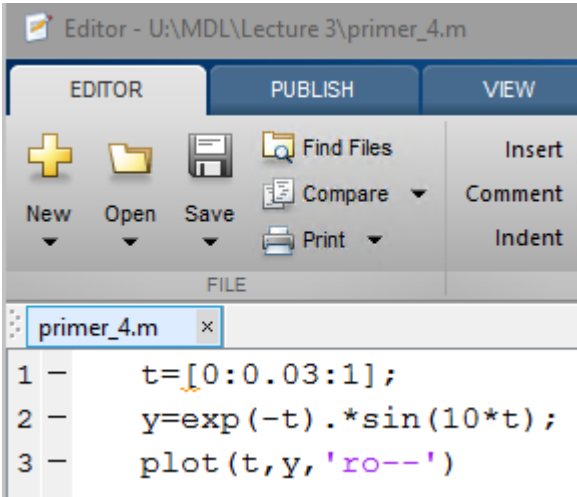


```
Editor - U:\MDL\Lecture 3\primer_3.m
EDITOR PUBLISH VIEW
+ New + Open + Save Find Files Insert
Compare Comment
Print Indent
FILE EDIT
primer_3.m
1 - t = [0:0.01:2];
2 - y = exp(-t).*sin(10*t);
3 - x = 100 * sin(50*t);
4 - %%plot(t,x,t,y);
5 - plotyy(t,x,t,y);
```

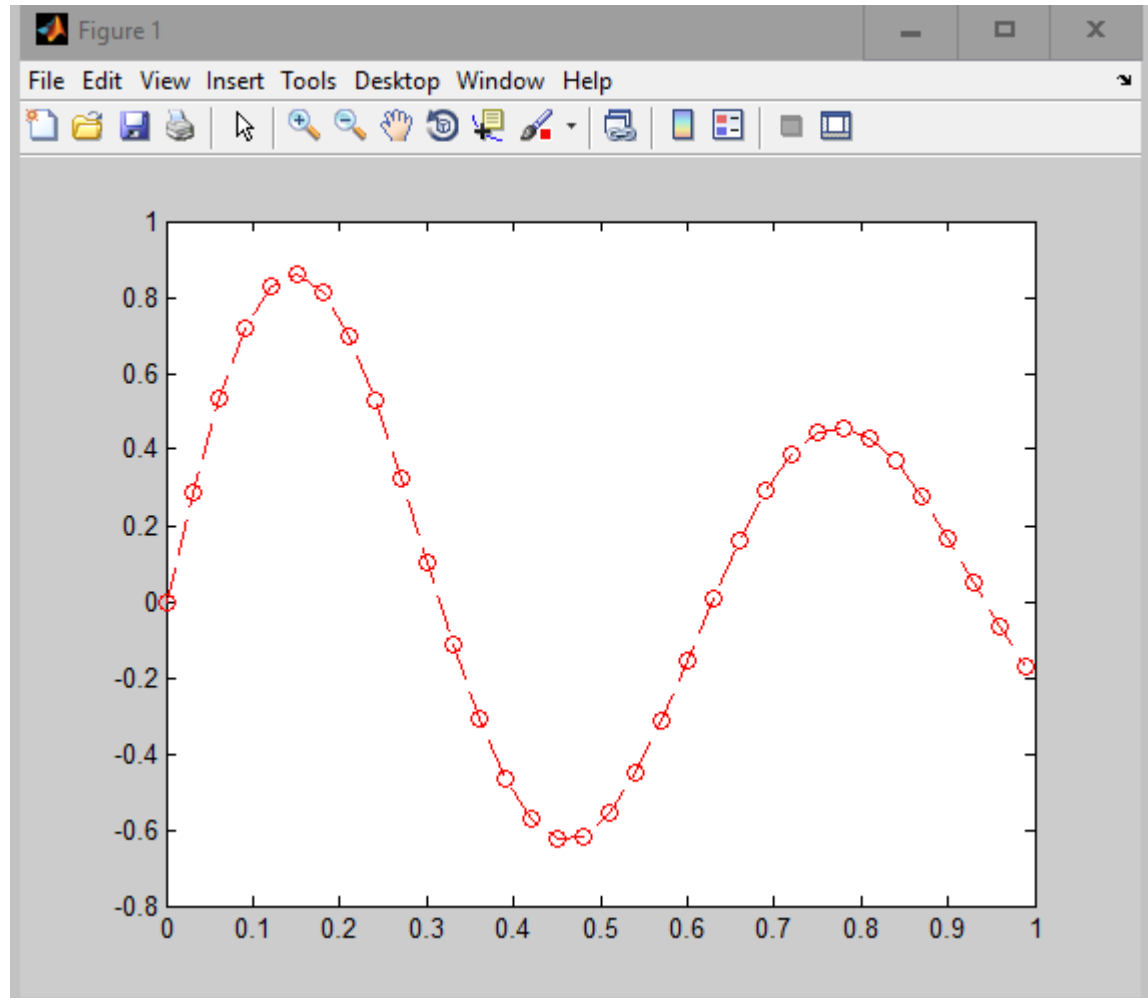


## 3.2.1 Графики в линейном масштабе

Построим график функции  $y(t) = e^{-t} \cdot \sin(10 \cdot t)$  в виде штриховой линии красного цвета, помеченной маркером типа «кружок».



```
Editor - U:\MDL\Lecture 3\primer_4.m  
EDITOR PUBLISH VIEW  
New Open Save Find Files Compare Print Insert Comment Indent  
FILE  
primer_4.m x  
1 - t=[0:0.03:1];  
2 - y=exp(-t).*sin(10*t);  
3 - plot(t,y,'ro--')
```





## 3.2.1 Графики в линейном масштабе

Дополнительные параметры для определения свойств линий графиков

Цвет		Тип маркера		Стиль линии	
1		2		3	
y	желтый	·	точка	-	сплошная
m	розовый	o	кружок	:	пунктирная
c	голубой	x	крестик	-.	штрих-пунктирная
r	красный	+	знак плюс	--	штриховая
g	зеленый	s	квадрат		
b	синий	d	ромб		
w	белый	v	треугольник вершиной вниз		
k	черный	^	треугольник вершиной вверх		
		<	треугольник вершиной влево		
		>	треугольник вершиной вправо		
		p	пятиконечная звезда		
		h	шестиконечная звезда		

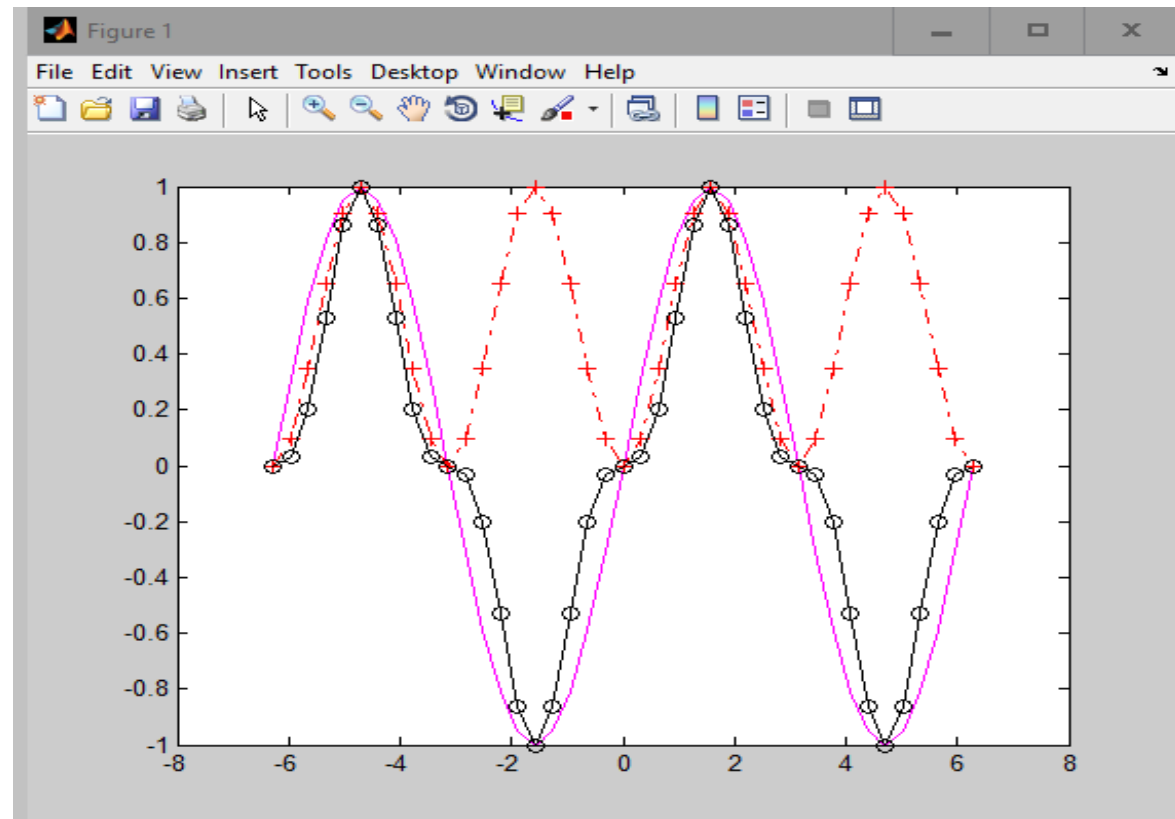
## 3.2.1 Графики в линейном масштабе

Построим график функции  $y_1(t) = \sin(t)$ ,  $y_2(t) = \sin^2(t)$  и  $y_3(t) = \sin^3(t)$  при изменении  $t$  в интервале  $[-2\pi, 2\pi]$  с шагом  $0,1\pi$ .

```

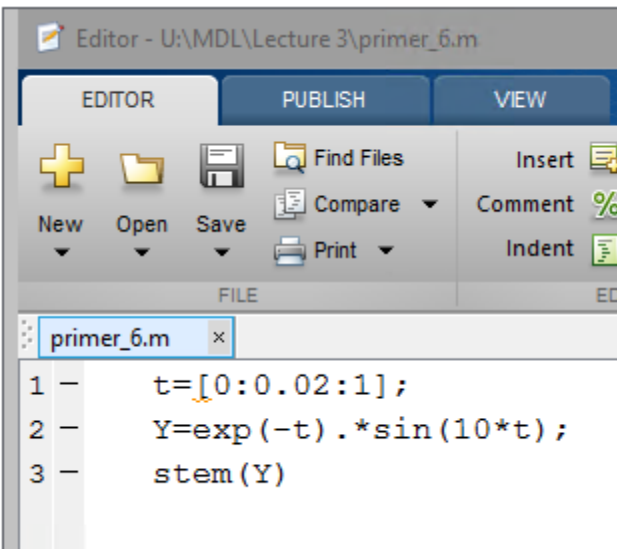
1 - t=[-2*pi:0.1*pi:2*pi];
2 - y1=sin(t);
3 - y2=sin(t).^2;
4 - y3=sin(t).^3;
5 - plot(t,y1,'-m',t,y2,'-+r',t,y3,'-ok')

```

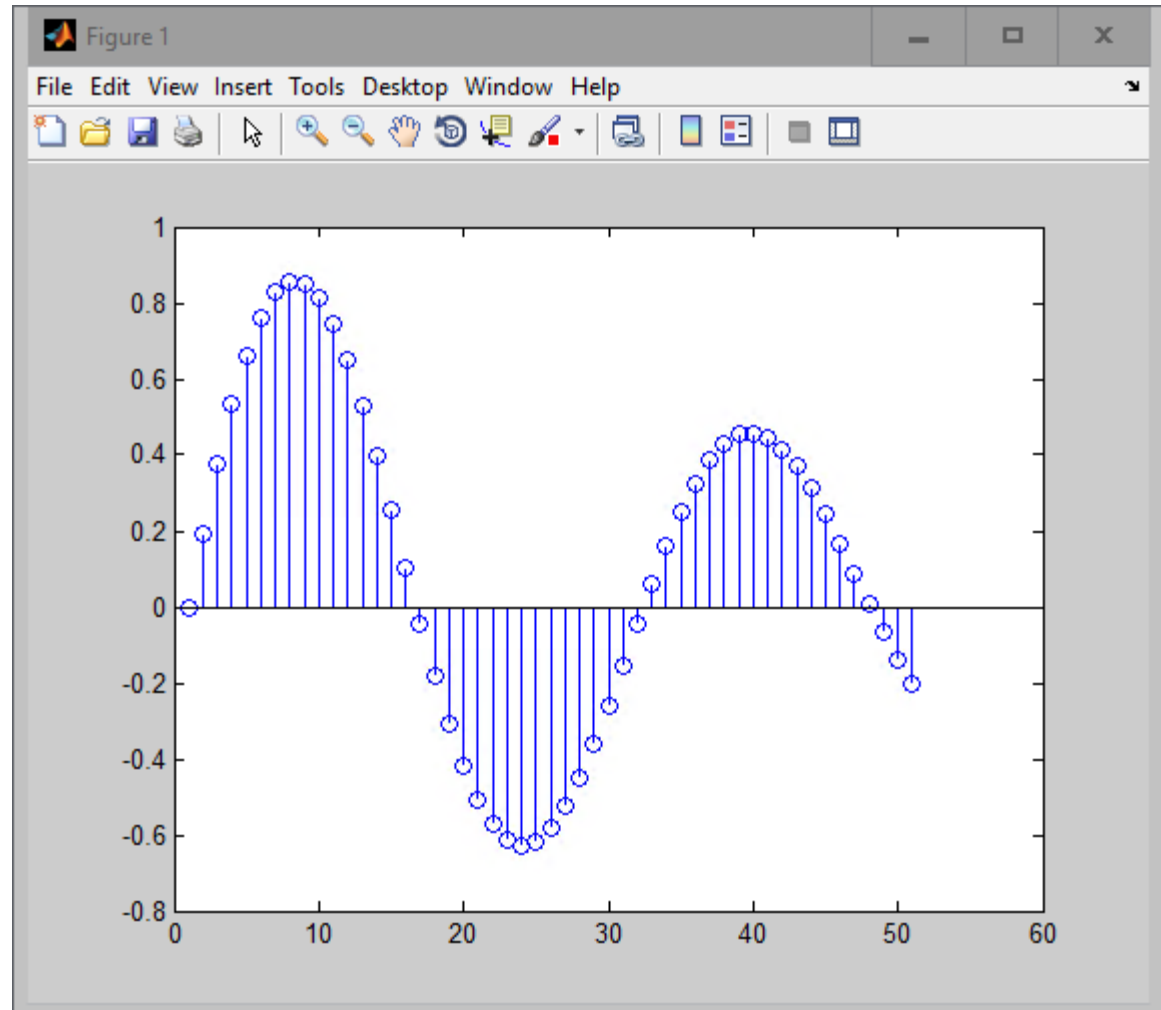


## 3.2.2 Графики в виде дискретных отсчетов

Построим график функции  $y(t) = e^{-t} \cdot \sin(10 \cdot t)$ , где аргумент  $t$  меняется от 0 до 1 с шагом 0,02. Функцию  $y$  зададим как вектор. Воспользуемся функцией `stem(Y)`.

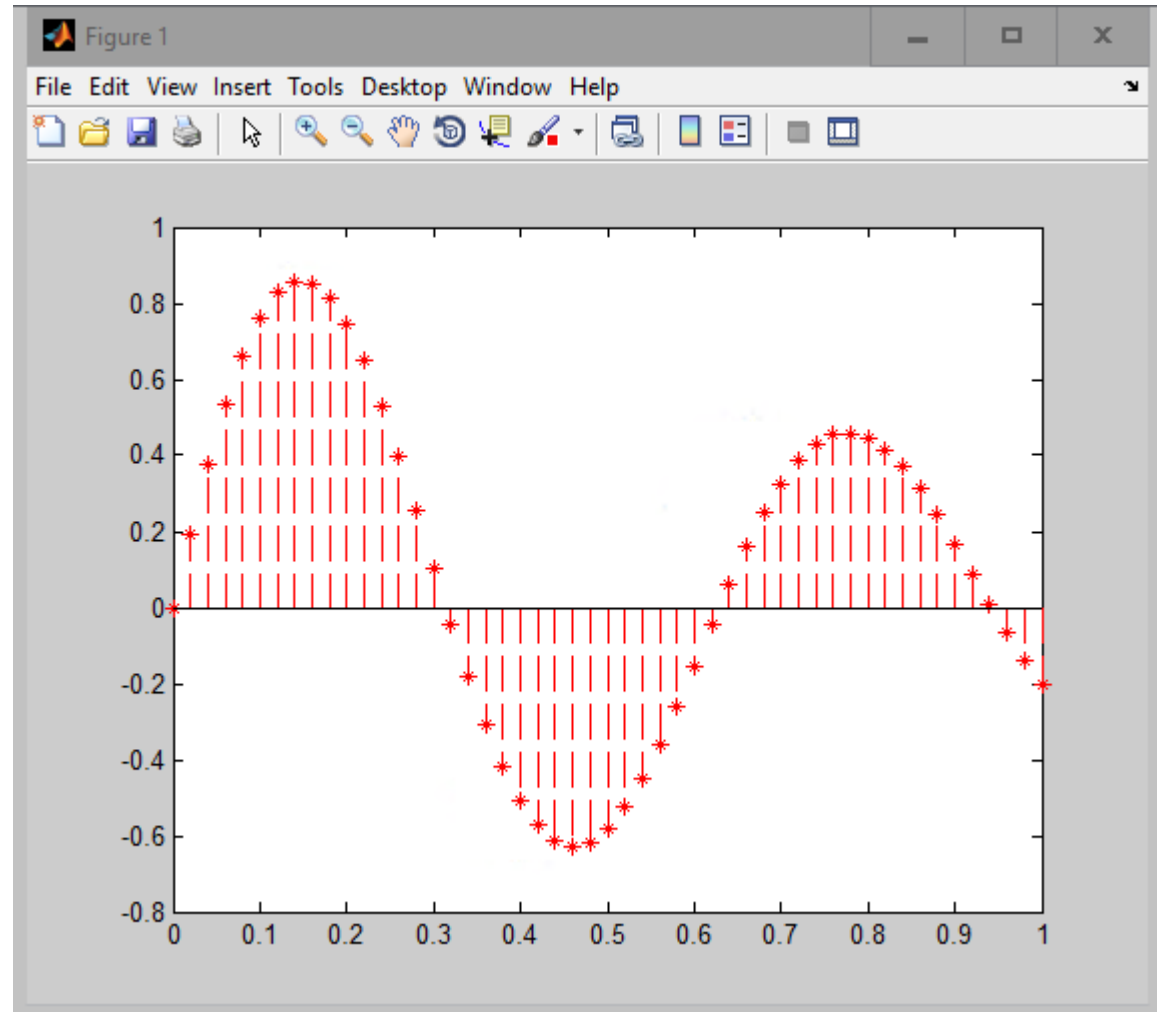


```
Editor - U:\MDL\Lecture 3\primer_6.m
EDITOR PUBLISH VIEW
+ New Open Save Find Files Compare Print Insert Comment Indent
primer_6.m x
1 - t=[0:0.02:1];
2 - Y=exp(-t).*sin(10*t);
3 - stem(Y)
```



## 3.2.2 Графики в виде дискретных отсчетов

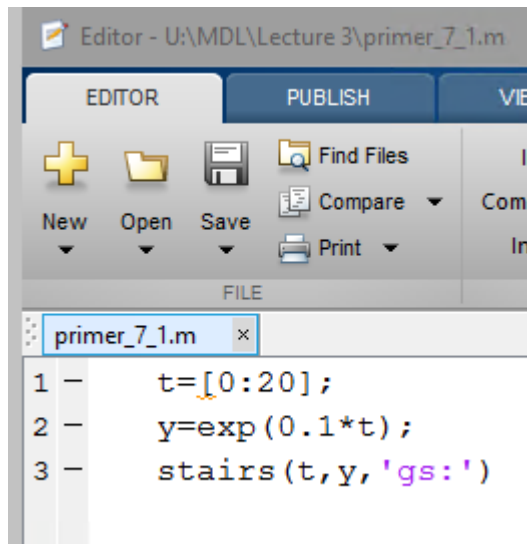
```
Editor - U:\MDL\Lecture 3\primer_7.m
EDITOR PUBLISH VIEW
+ New Open Save Find Files Compare Print Insert Comment Indent
primer_7.m
1 - t=[0:0.02:1];
2 - Y=exp(-t).*sin(10*t);
3 - stem(t,Y,'r*--')
```



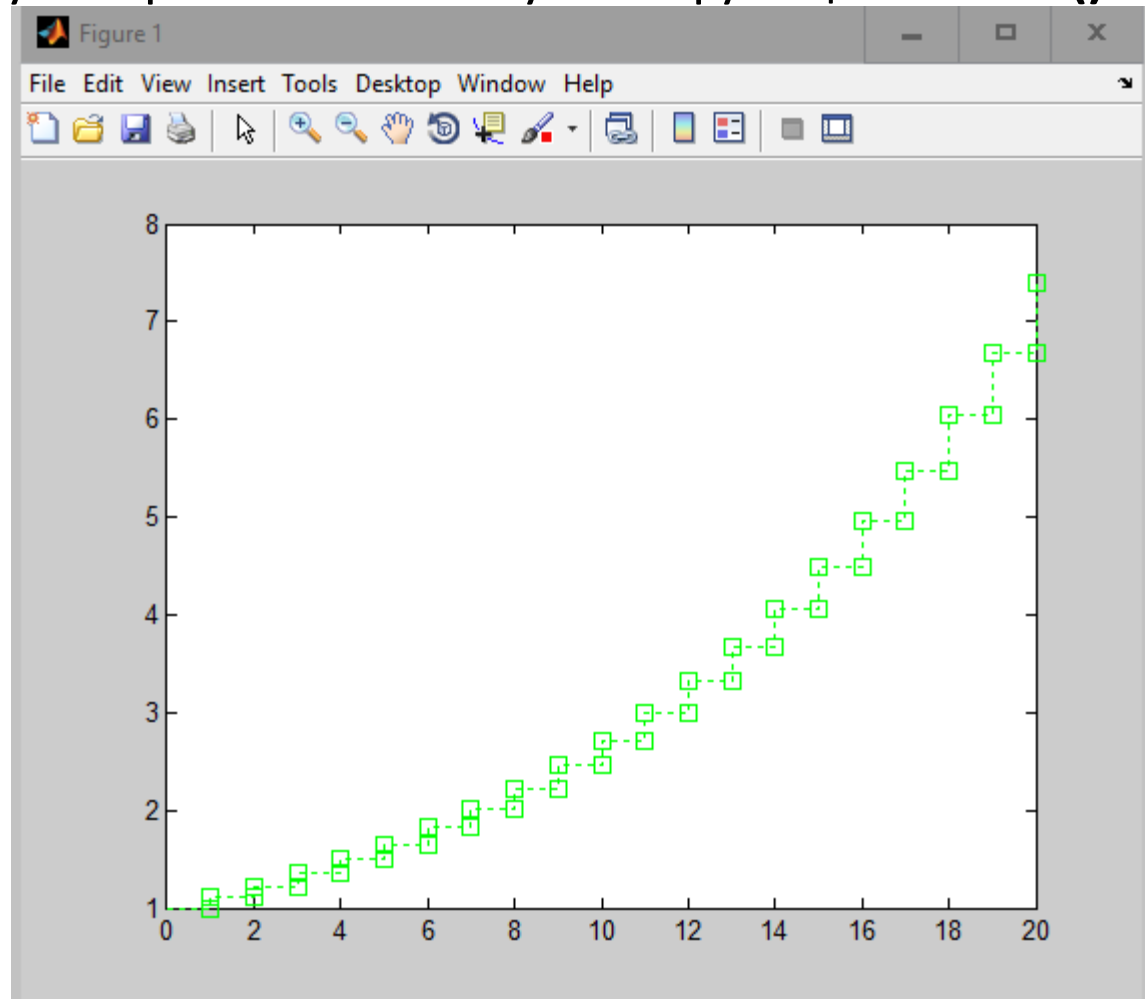
### 3.2.3 Лестничные графики

Построим лестничный график функции  $y(t)=e^{0,1t}$ , где аргумент  $t$  меняется от 0 до 20 с шагом 1.

Дополнительного зададим параметры: цвет – зеленый, тип маркера – квадрат, стиль линии – пунктирная. Воспользуемся функцией `stairs()`.



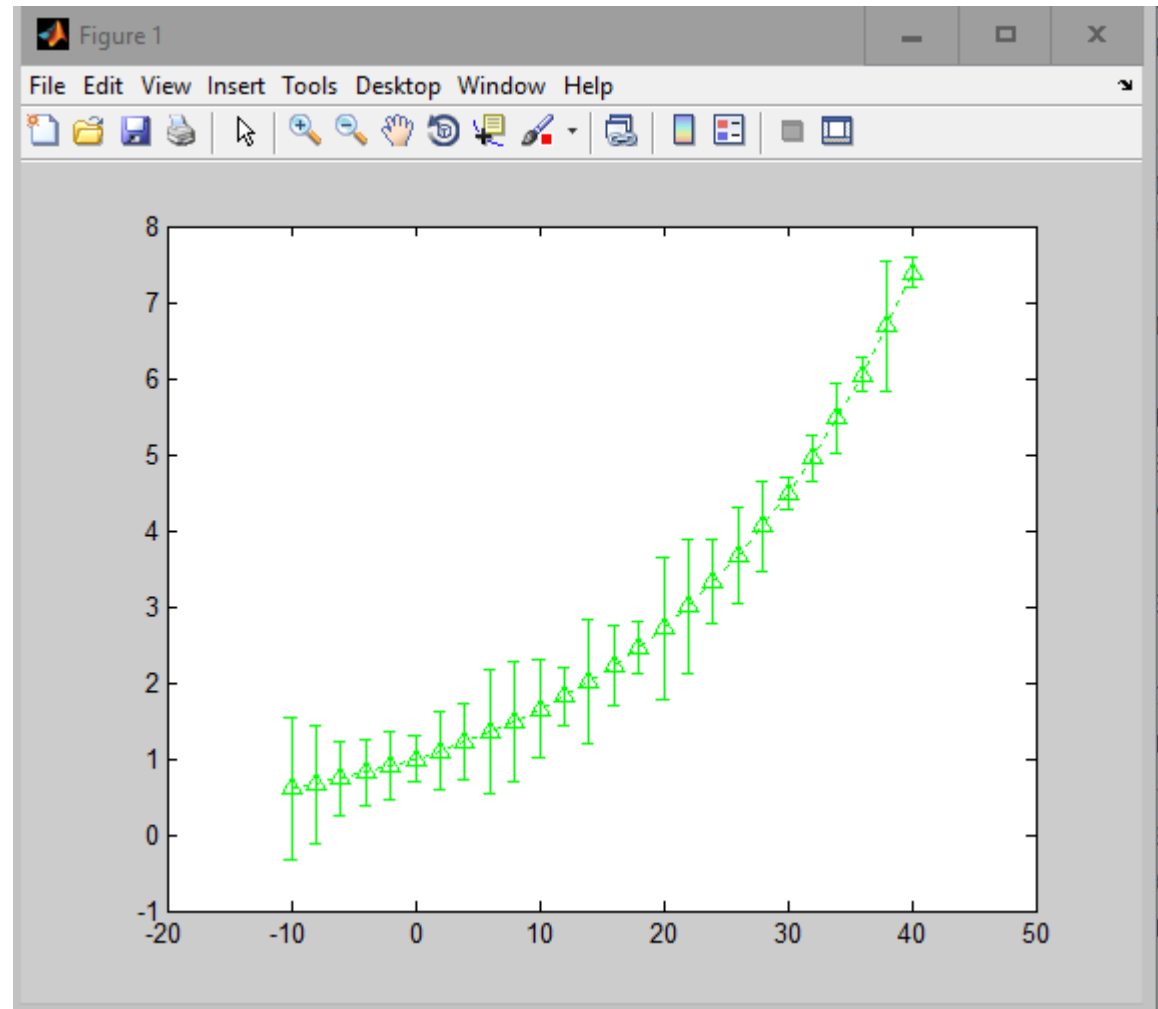
```
Editor - U:\MDL\Lecture 3\primer_7_1.m
EDITOR PUBLISH VIEW
+ New Open Save Find Files Compare Print
FILE
primer_7_1.m
1 - t=[0:20];
2 - y=exp(0.1*t);
3 - stairs(t,y,'gs:')
```



## 3.2.4 Графики с указанием погрешности

Построим лестничный график функции  $y(t)=e^{0,05t}$ , где аргумент  $t$  меняется от -10 до 40 с шагом 2. Ошибку  $E$  зададим последовательностью случайных чисел. Воспользуемся функциями **errorbar()**, **rand()**, **size()**.

```
Editor - U:\MDL\Lecture 3\primer_7_2.m
EDITOR PUBLISH VIEW
New Open Save Find Files Compare Print Insert Comment Indent
FILE EDIT
primer_7_2.m
1 - t=[-10:2:40];
2 - y=exp(0.05*t);
3 - % size() определяет
4 - % размер массива
5 - % сгенерированных чисел
6 - E=rand(size(10*t));
7 - errorbar(t,y,E,'g^:')
```



## 3.2.5 Графики в логарифмическом формате

Для построения графиков в логарифмическом и полул로그арифмическом масштабе служат следующие функции:

`Loglog()` – логарифмический масштаб по обеим осям;

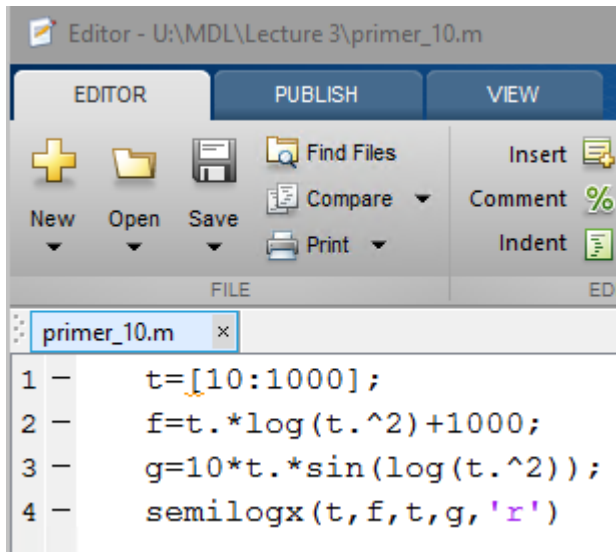
`semilogx()` – логарифмический масштаб по оси абсцисс;

`semilogy()` – логарифмический масштаб по оси ординат.

Аргументы этих функций (все, что стоит внутри скобок) формируются по тем же правилам, что и в функции `plot()`.

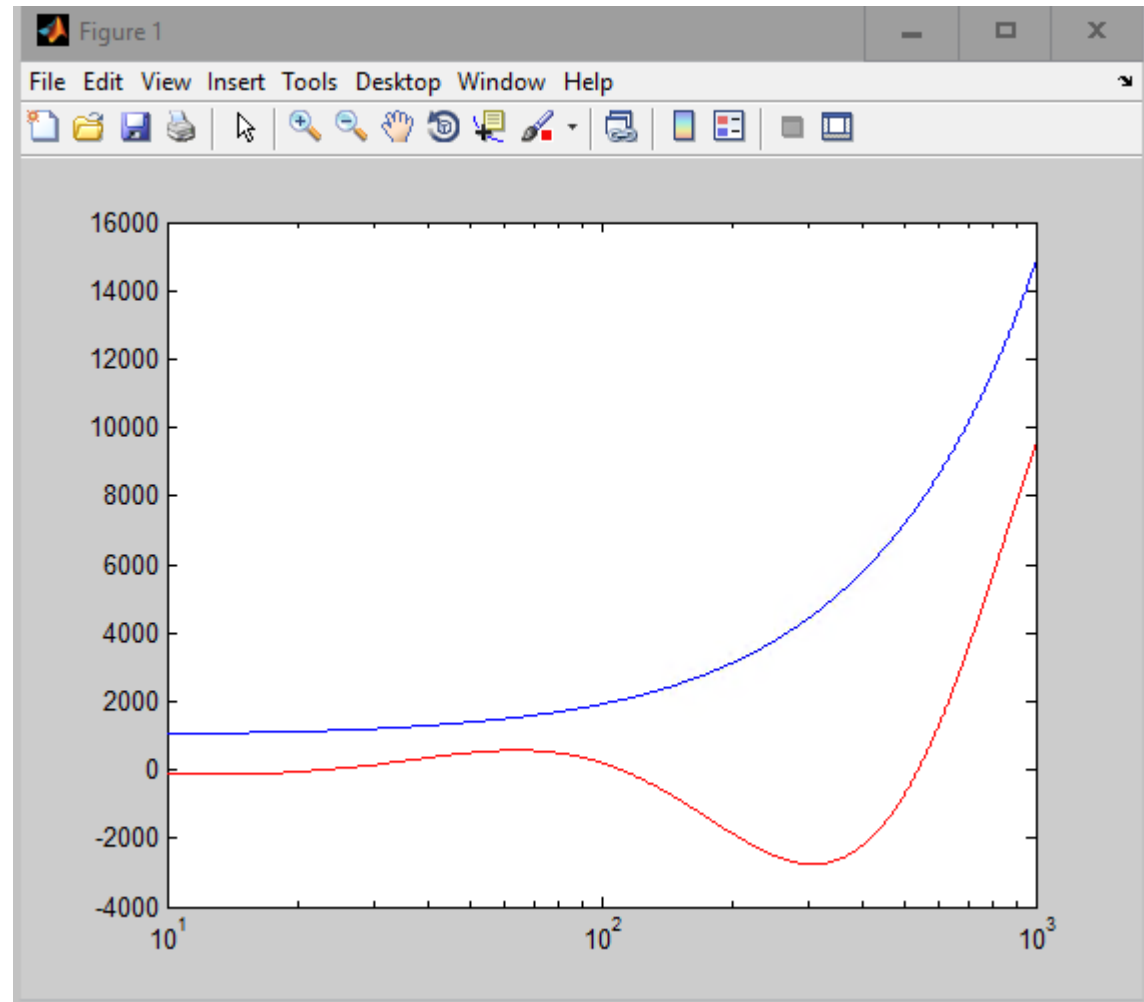
## 3.2.5 Графики в логарифмическом формате

Построим в одном окне в полулогарифмическом масштабе с использованием функции **semilogx()** графики зависимостей  $f(t)=t \cdot \log(t^2)+1000$  и  $g(t)=10 \cdot t \cdot \sin(\log(t^2))$  при условии, что  $t$  меняется на интервале  $[10, 1000]$  с шагом 1. График функции  $g(t)$  выделим красным цветом.



```

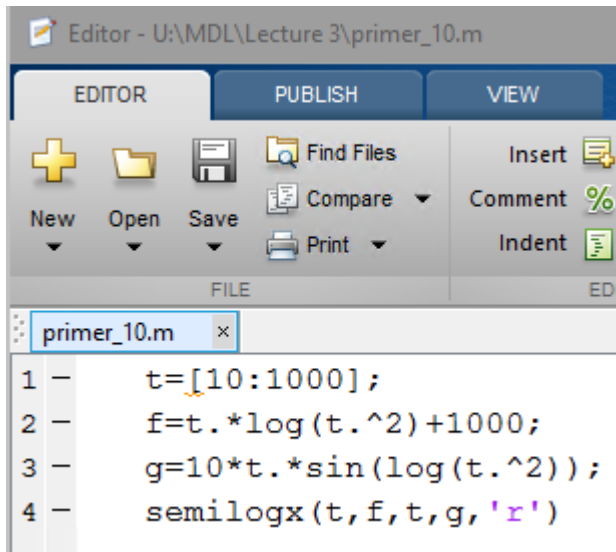
1 - t=[10:1000];
2 - f=t.*log(t.^2)+1000;
3 - g=10*t.*sin(log(t.^2));
4 - semilogx(t, f, t, g, 'r')
  
```





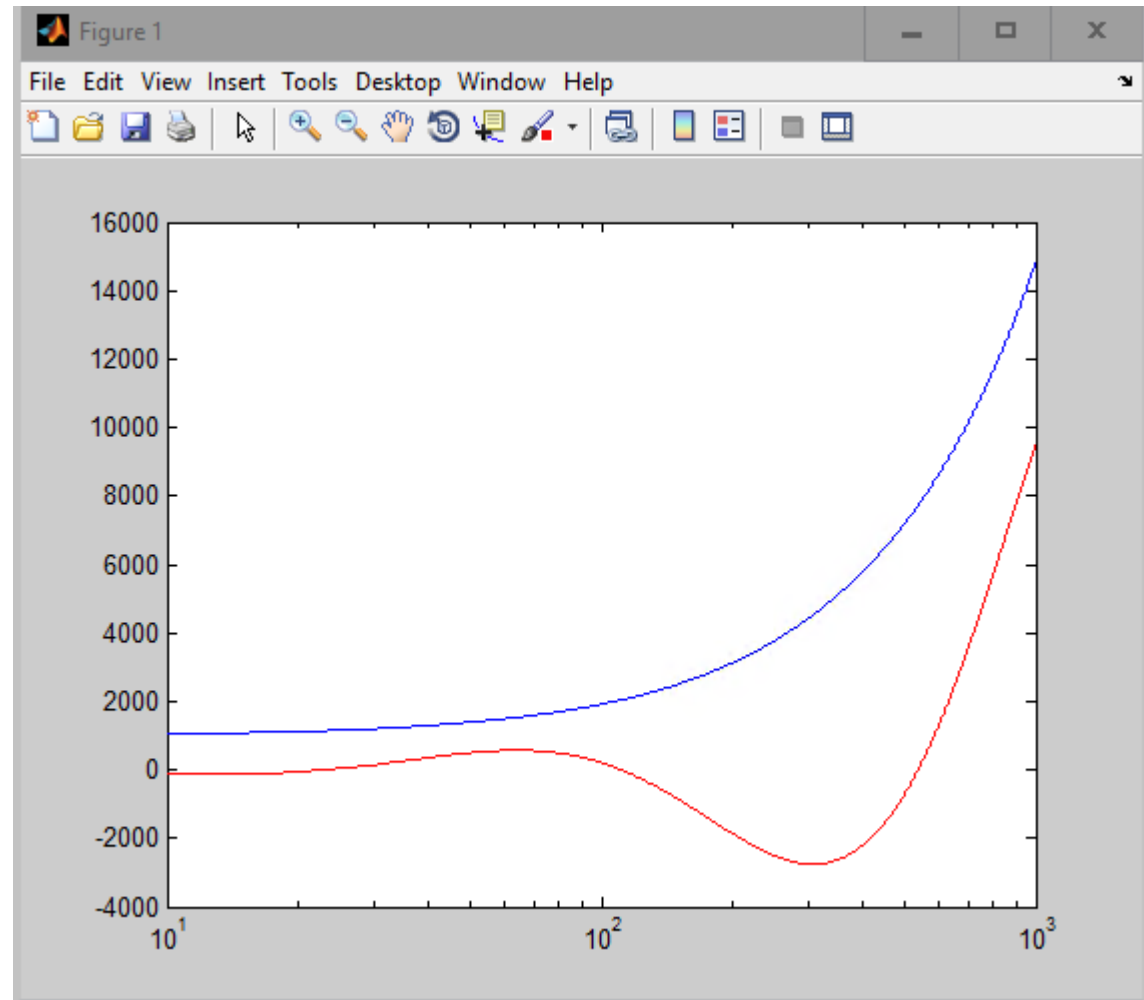
## 3.2.5 Графики в логарифмическом формате

Построим в одном окне в полулогарифмическом масштабе с использованием функции **semilogx()** графики зависимостей  $f(t)=t \cdot \log(t^2)+1000$  и  $g(t)=10 \cdot t \cdot \sin(\log(t^2))$  при условии, что  $t$  меняется на интервале  $[10, 1000]$  с шагом 1. График функции  $g(t)$  выделим красным цветом.



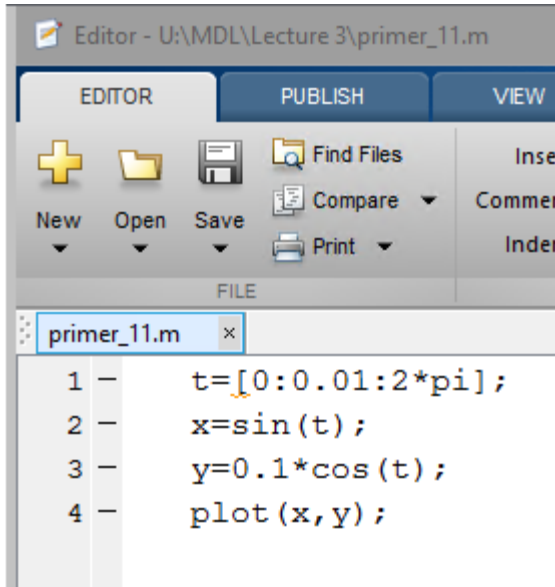
```

1 - t=[10:1000];
2 - f=t.*log(t.^2)+1000;
3 - g=10*t.*sin(log(t.^2));
4 - semilogx(t, f, t, g, 'r')
  
```

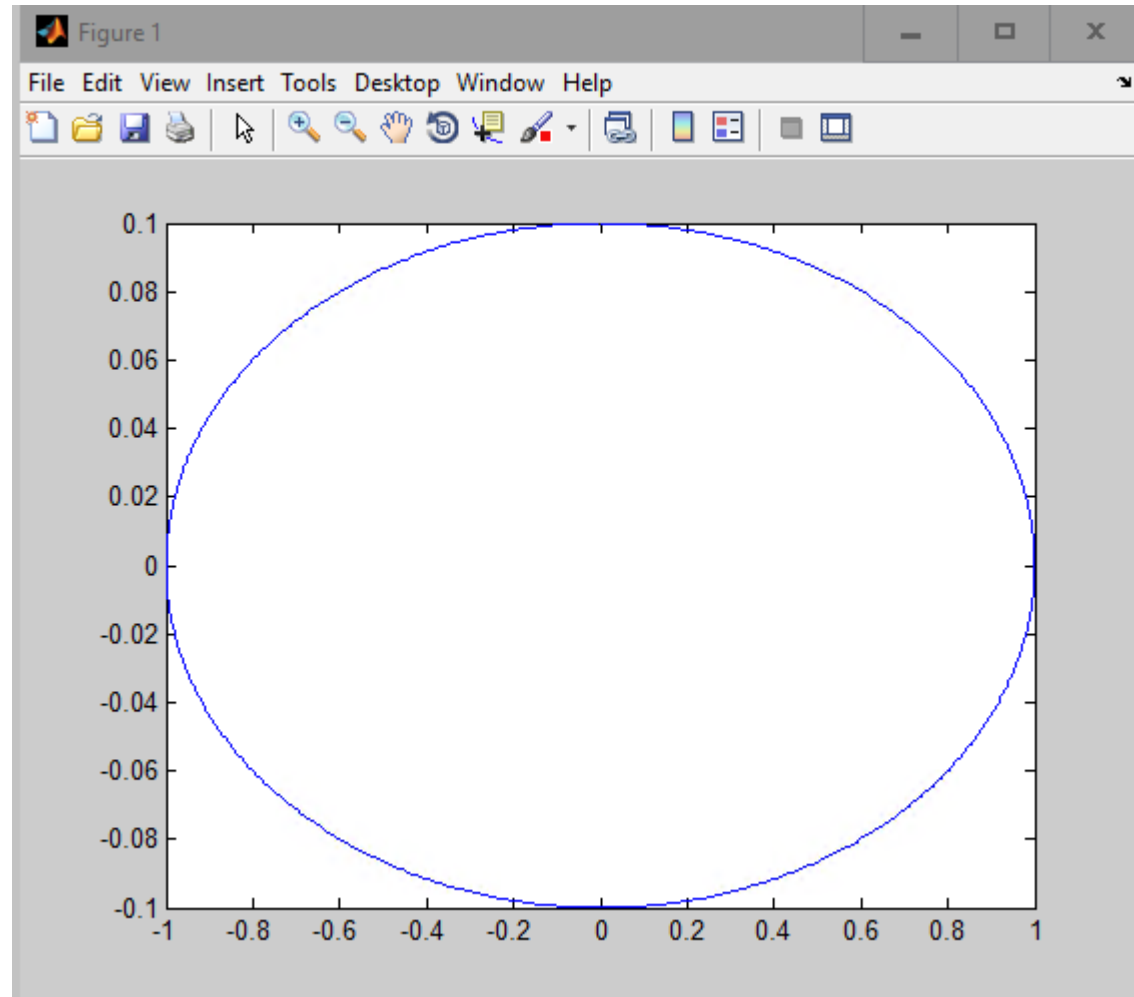


## 3.2.6 Графики параметрических функций

Используя в качестве исходных функций  $x(t)=\sin(t)$  и  $y(t)=0,1\cdot\cos(t)$ , построим график параметрической функции  $y(x)$  при условии, что  $t$  изменяется на интервале  $[0, 2\pi]$  с шагом  $0,01$ .



```
Editor - U:\MDL\Lecture 3\primer_11.m
EDITOR PUBLISH VIEW
+ Find Files Inse
New Open Save Compare Commer
Print Index
FILE
primer_11.m x
1 - t=[0:0.01:2*pi];
2 - x=sin(t);
3 - y=0.1*cos(t);
4 - plot(x,y);
```



## 3.3 Построение трехмерных графиков

Под 3D-графикой понимают построение графической зависимости функции двух переменных  $z(x, y)$ . График такой функции представляет собой изображение некоторой поверхности в трехмерном пространстве и строится с использованием аксонометрического метода.

Для графической визуализации функции двух переменных следует:

а) сформировать матрицу  $[x, y]$  с координатами узлов сетки на прямоугольной области определения функции. Матрица с координатами узлов сетки генерируется с помощью команды **meshgrid(x, y)**. Аргументами  $x$  и  $y$  являются векторы, элементы которых задают координаты узлов прямоугольной сетки. Если область определения функции квадрат, то в команде **meshgrid()** можно использовать один аргумент;

б) вычислить значения функции в узлах сетки;

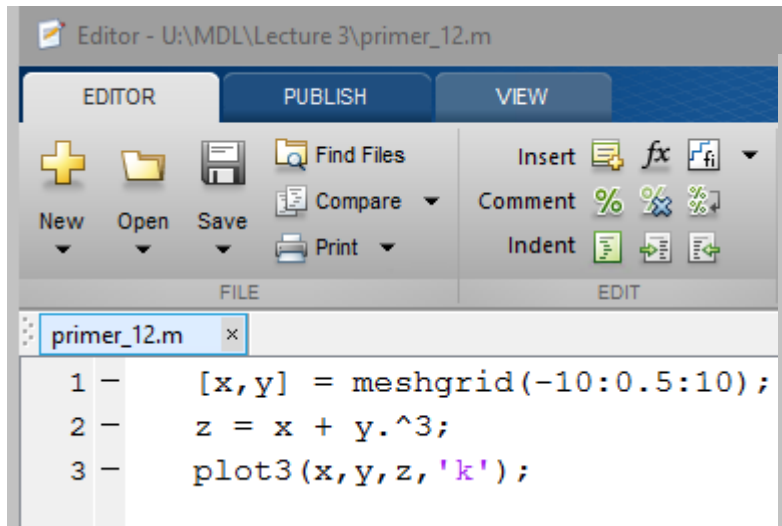
в) использовать для вывода графика одну из функций MatLab;

г) используя команды оформления графика, нанести дополнительную информацию.

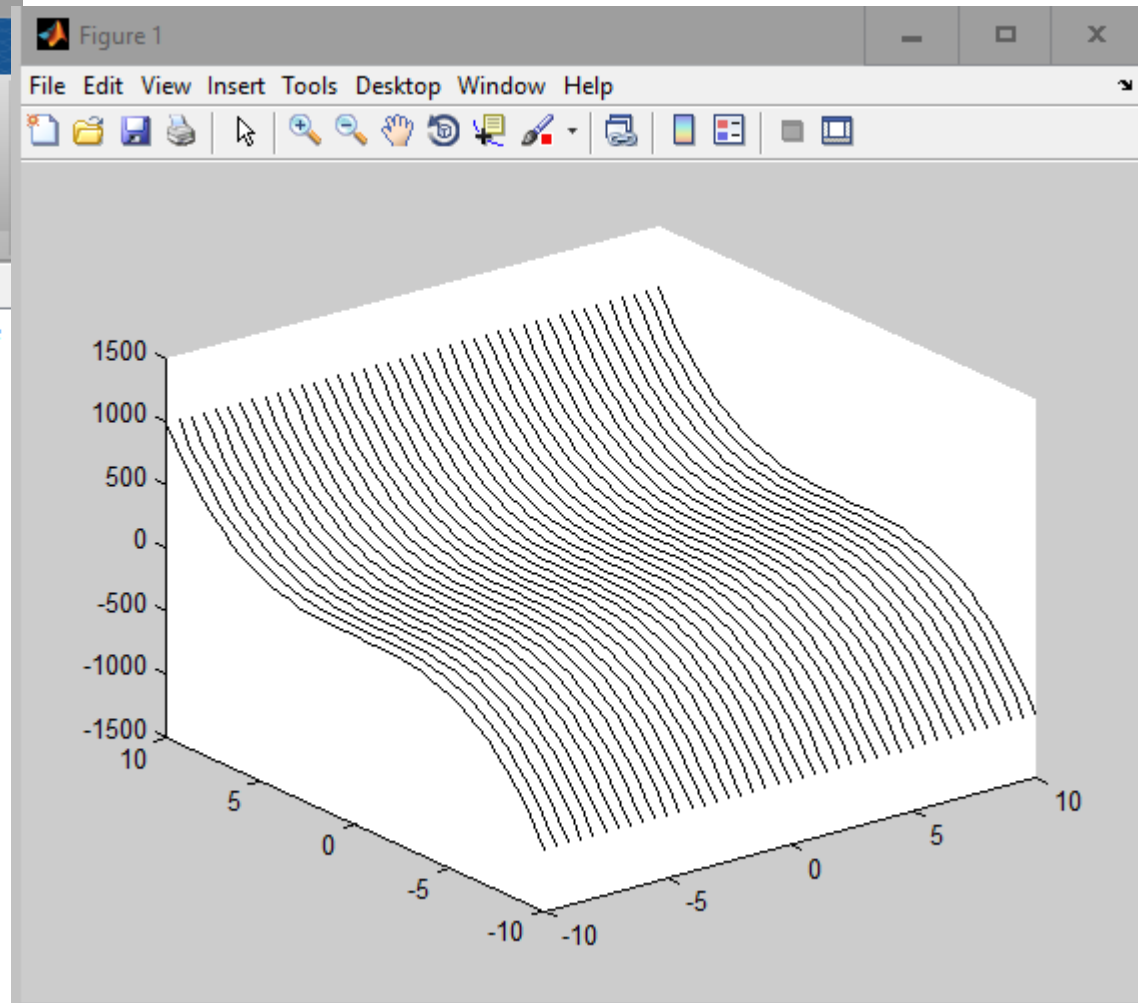
Необходимо учитывать, что главным условием для отображения графических зависимостей функции  $z(x, y)$  является понимание того, как в среде MatLab осуществляются поэлементные операции с векторами.

### 3.3.1 Линейчатые поверхности

Построим график функции  $z(x, y) = x + y^3$ , где переменные  $x$  и  $y$  изменяются на интервале  $[-10, 10]$  с шагом  $0,5$ . Для построения графика используем непрерывные линии черного цвета.



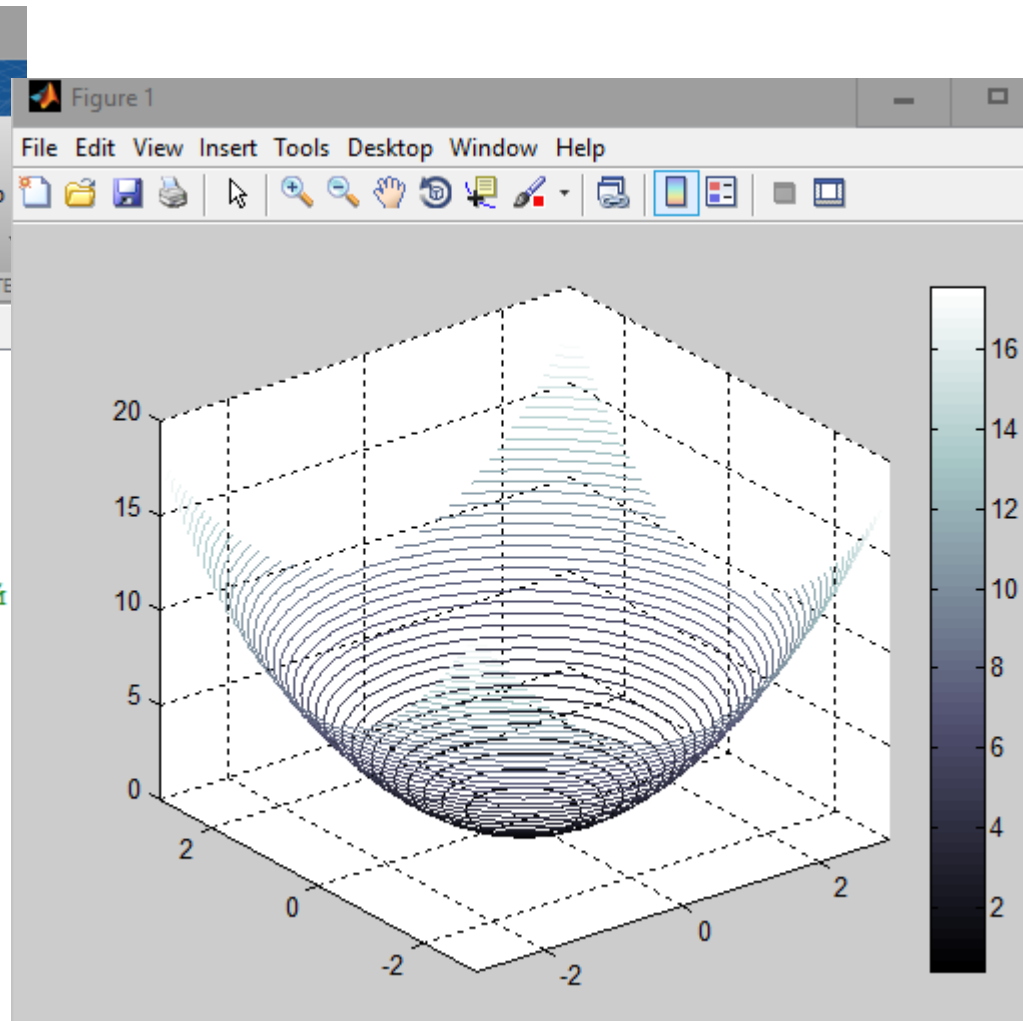
```
Editor - U:\MDL\Lecture 3\primer_12.m  
EDITOR PUBLISH VIEW  
+ Find Files Insert fx fi  
New Open Save Compare Comment % %  
Print Indent  
FILE EDIT  
primer_12.m x  
1 - [x,y] = meshgrid(-10:0.5:10);  
2 - z = x + y.^3;  
3 - plot3(x,y,z,'k');
```



### 3.3.1 Линейчатые поверхности

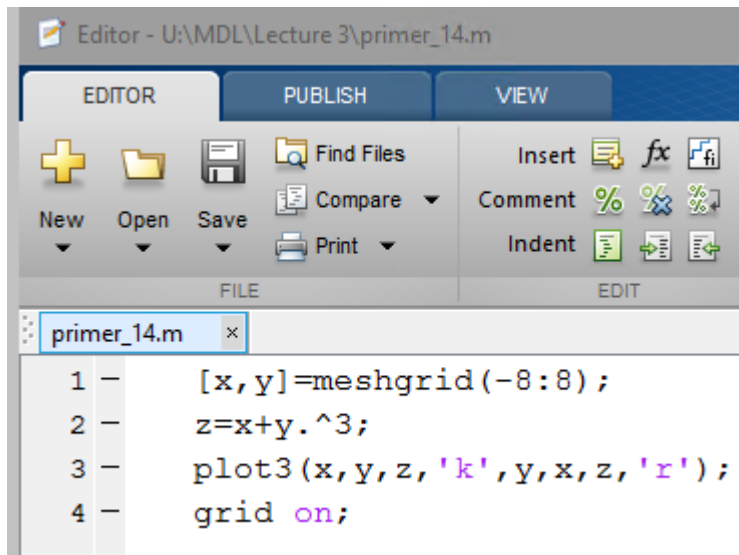
С использованием функции `contour3()` построим линейчатую поверхность функции  $z = x^2 + y^2$ . Переменные  $x$  и  $y$  изменяются на интервале  $[-3, 3]$  с шагом  $0,1$ . Количество линий на графике – 40.

```
Editor - U:\MDL\Lecture 3\primer_13.m
EDITOR PUBLISH VIEW
New Open Save Find Files Compare Print
Insert fx fx
Comment % %
Indent % %
FILE EDIT NAVIGATE
primer_13.m x
1 - [x,y]=meshgrid(-3:0.1:3);
2 - z=x.^2+y.^2;
3 - contour3(x,y,z,40);
4 - colormap(bone); % задает окраску
5 - colorbar; % выводит рядом с
6 - % графиком столбик, устанавливающий
7 - % соответствие между
8 - % цветом поверхности
9 - % и величиной функции
```

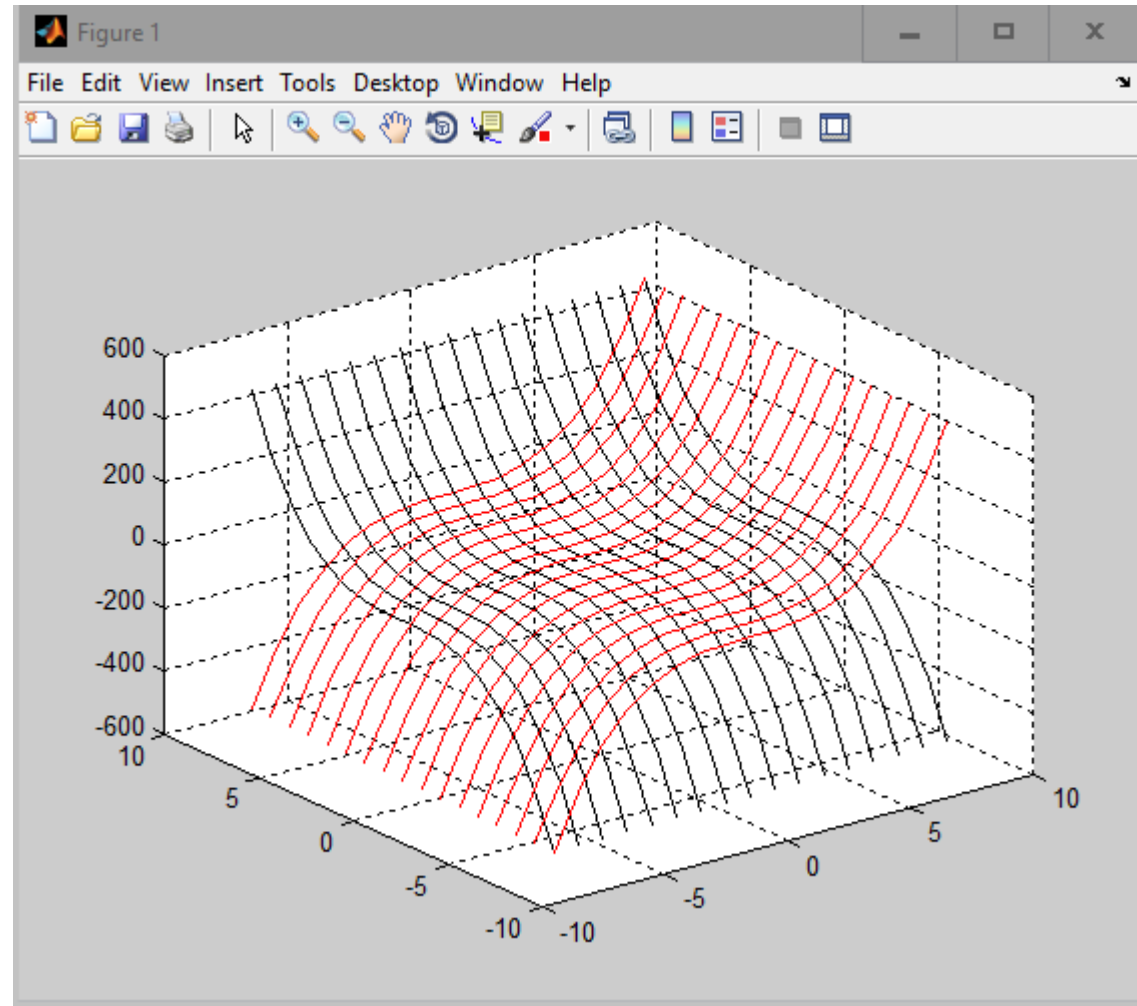


## 3.3.2 Каркасные поверхности

Построим график каркасной поверхности функции  $z(x, y) = x + y^3$ , где переменные  $x$  и  $y$  изменяются на интервале  $[-10, 10]$  с шагом 1. Для построения графика используем непрерывные линии. Линии, расположенные параллельно плоскости  $ZOY$ , закрасим в черный цвет, а линии, параллельные плоскости  $ZOX$  – в красный цвет. На плоскостях  $XOY$ ,  $ZOX$ ,  $ZOY$  сформируем сетку.

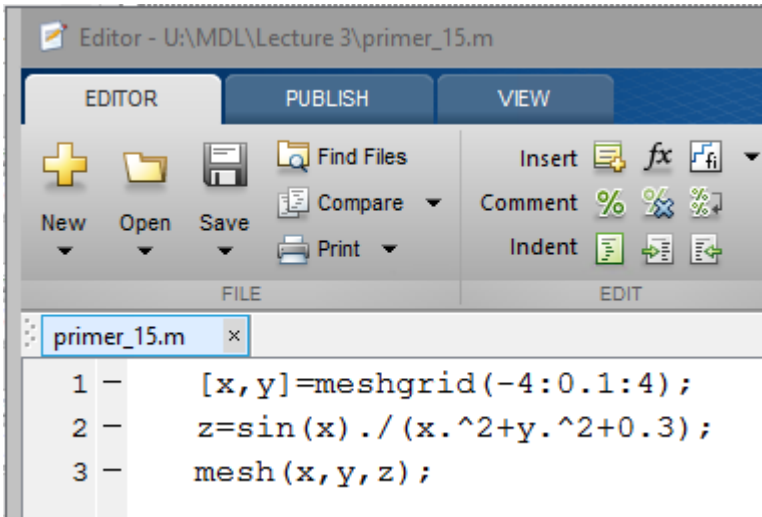


```
Editor - U:\MDL\Lecture 3\primer_14.m  
EDITOR PUBLISH VIEW  
New Open Save Find Files Compare Print  
Insert fx  
Comment %  
Indent  
FILE EDIT  
primer_14.m  
1 - [x,y]=meshgrid(-8:8);  
2 - z=x+y.^3;  
3 - plot3(x,y,z,'k',y,x,z,'r');  
4 - grid on;
```

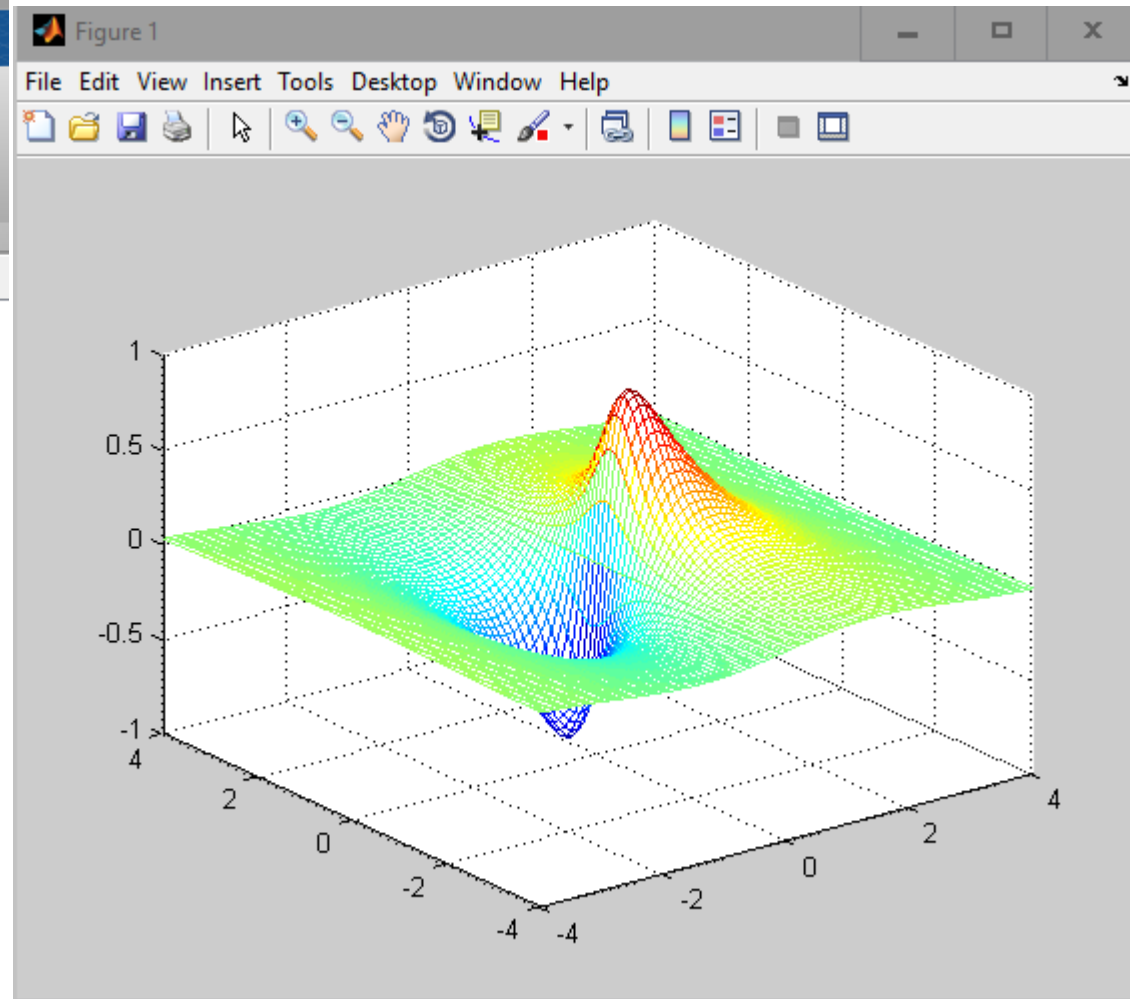


### 3.3.2 Каркасные поверхности

С помощью функции `mesh()` построим каркасную поверхность, заданную функцией  $z = \sin(x) / (x^2 + y^2 + 0,3)$ , где переменные  $x$  и  $y$  изменяются на интервале  $[-4, 4]$  с шагом  $0,1$ .

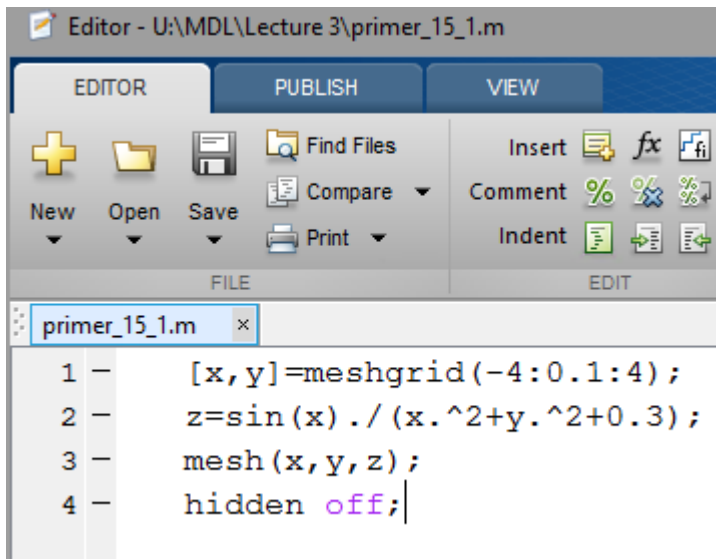


```
Editor - U:\MDL\Lecture 3\primer_15.m  
EDITOR PUBLISH VIEW  
New Open Save Find Files Compare Print  
Insert fx  
Comment %  
Indent  
FILE EDIT  
primer_15.m  
1 - [x,y]=meshgrid(-4:0.1:4);  
2 - z=sin(x)./(x.^2+y.^2+0.3);  
3 - mesh(x,y,z);
```

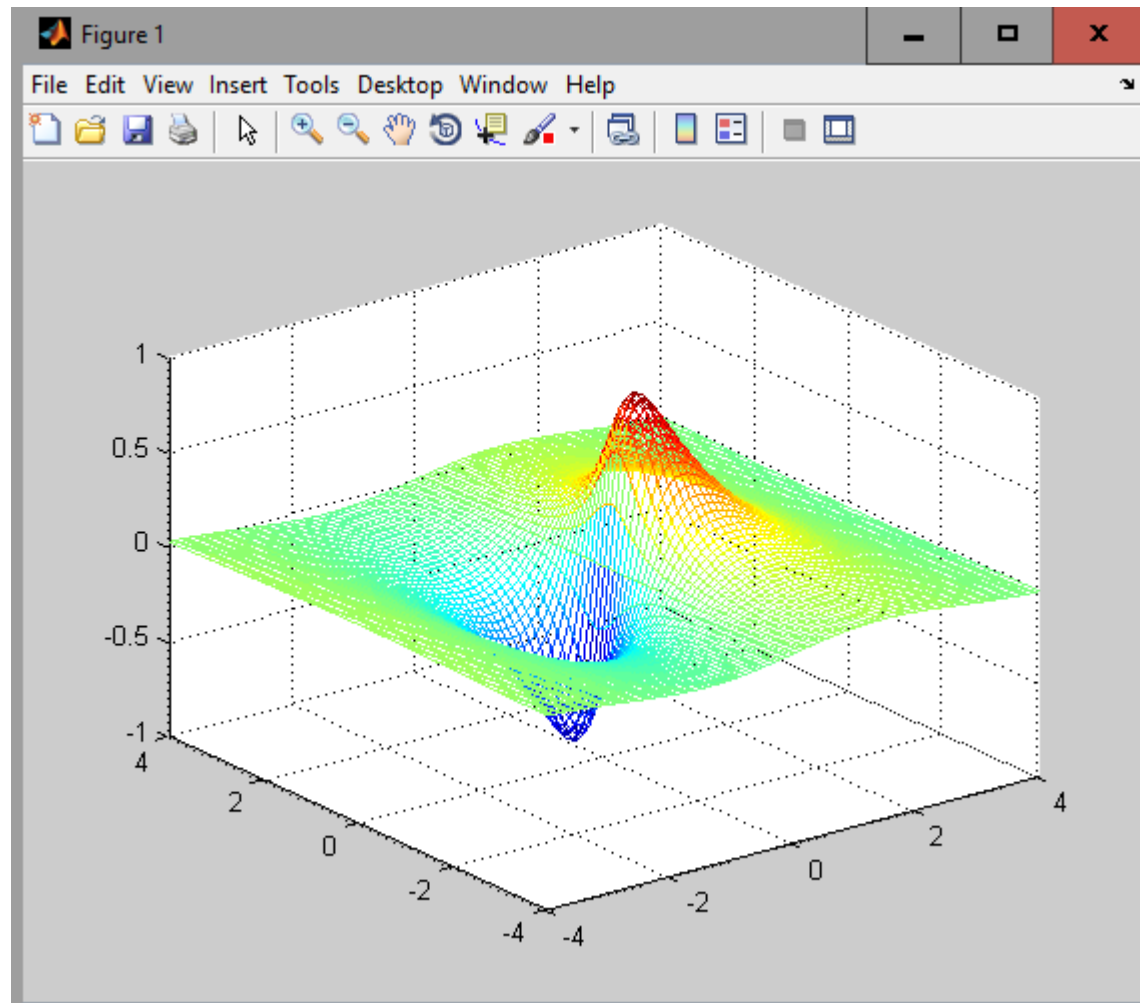


### 3.3.3 Прозрачная каркасная поверхность

Для построения прозрачной поверхности для случая на лайде 23 используем команду `hidden off`.



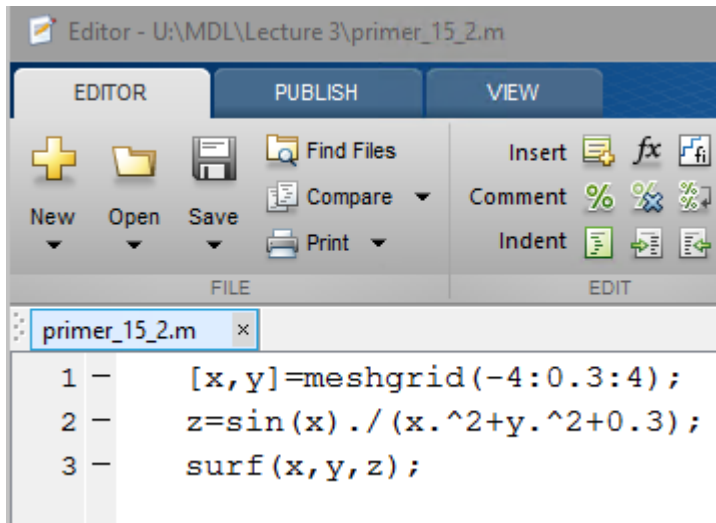
```
Editor - U:\MDL\Lecture 3\primer_15_1.m
EDITOR PUBLISH VIEW
New Open Save Find Files Compare Print Insert Comment Indent
[x, y]=meshgrid(-4:0.1:4);
z=sin(x) ./ (x.^2+y.^2+0.3);
mesh(x, y, z);
hidden off;
```





### 3.3.4 Каркасная поверхность залитая цветом

Используем функцию `surf()` для слайда 23.

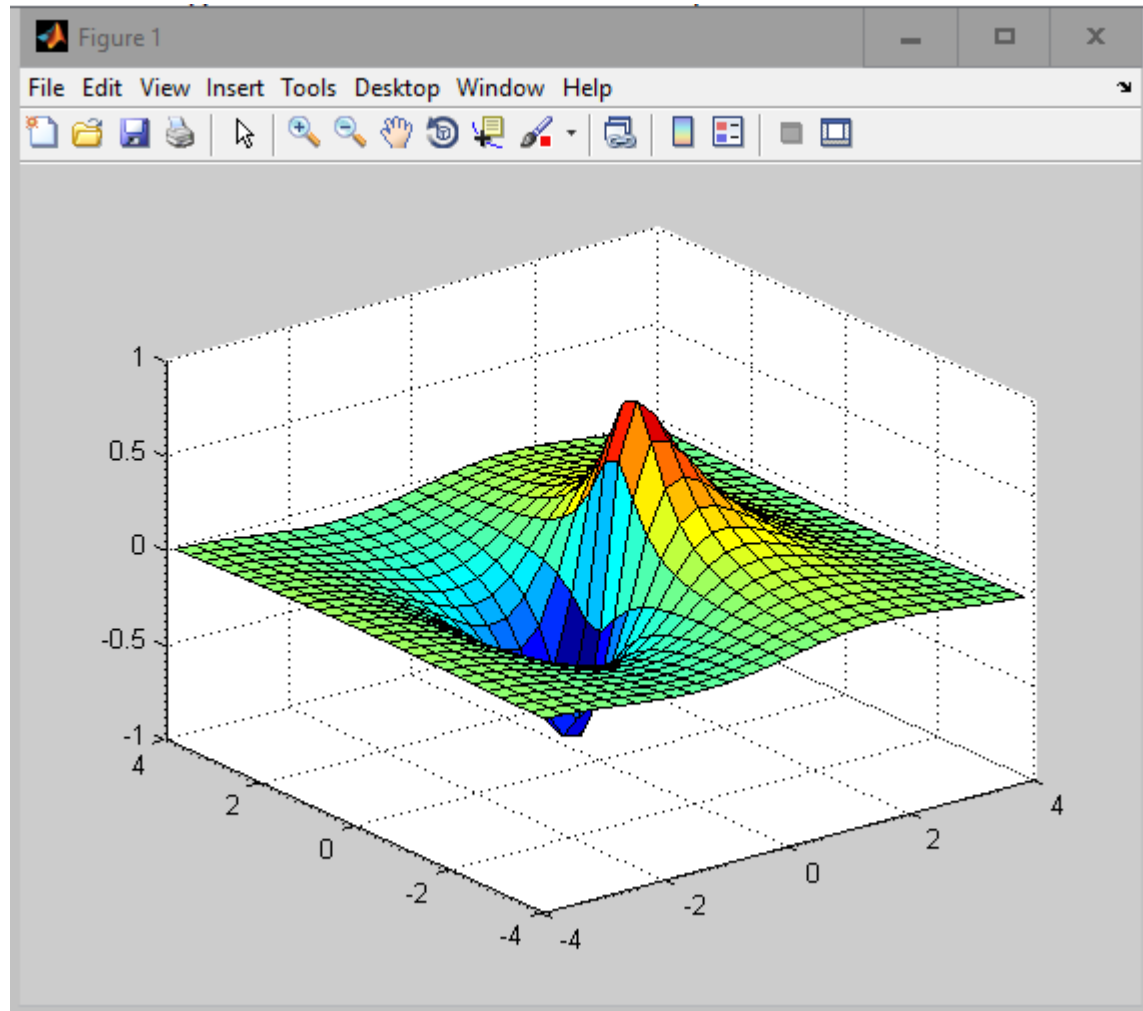


Editor - U:\MDL\Lecture 3\primer\_15\_2.m

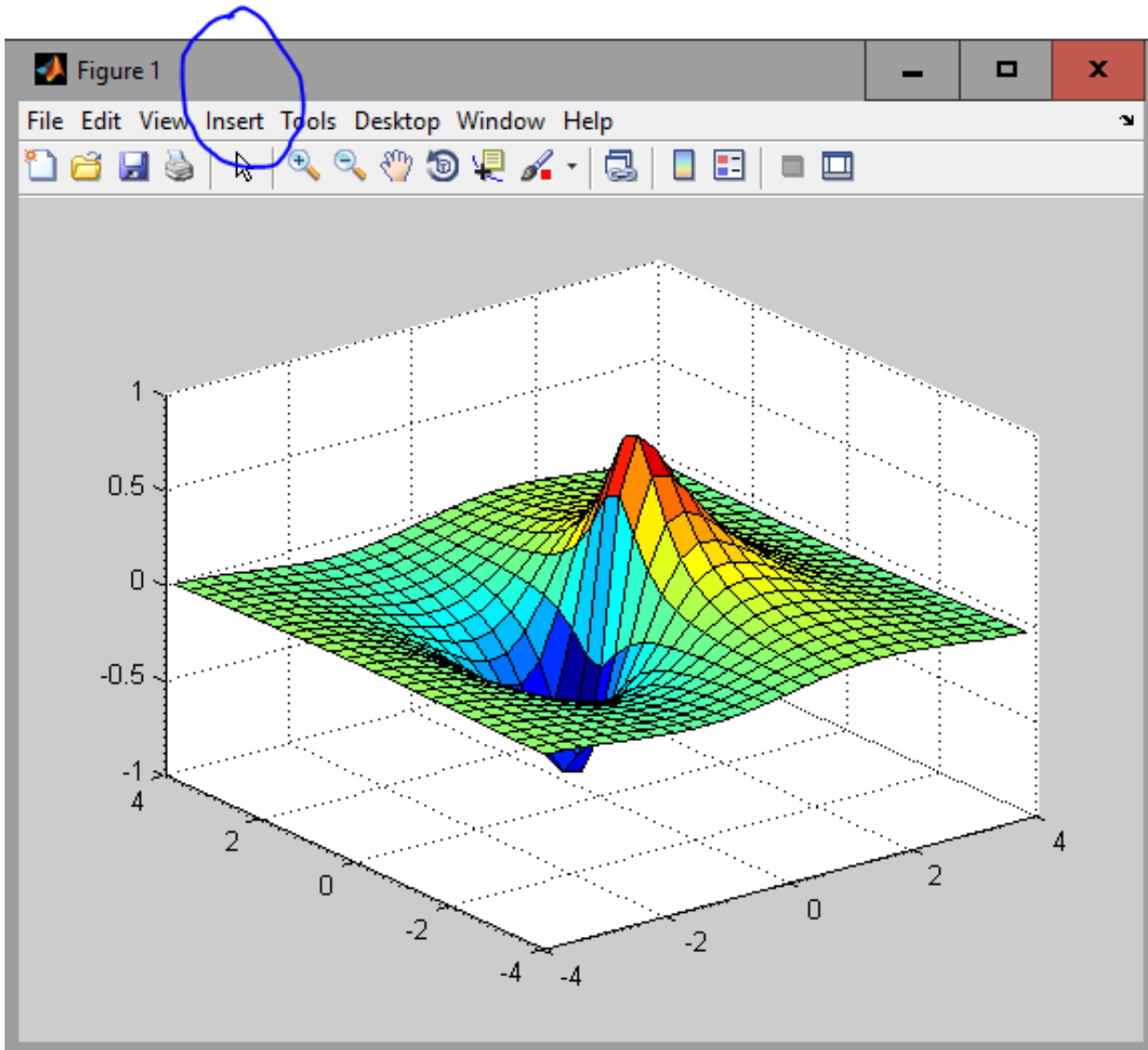
EDITOR PUBLISH VIEW

New Open Save Find Files Compare Print Insert Comment Indent

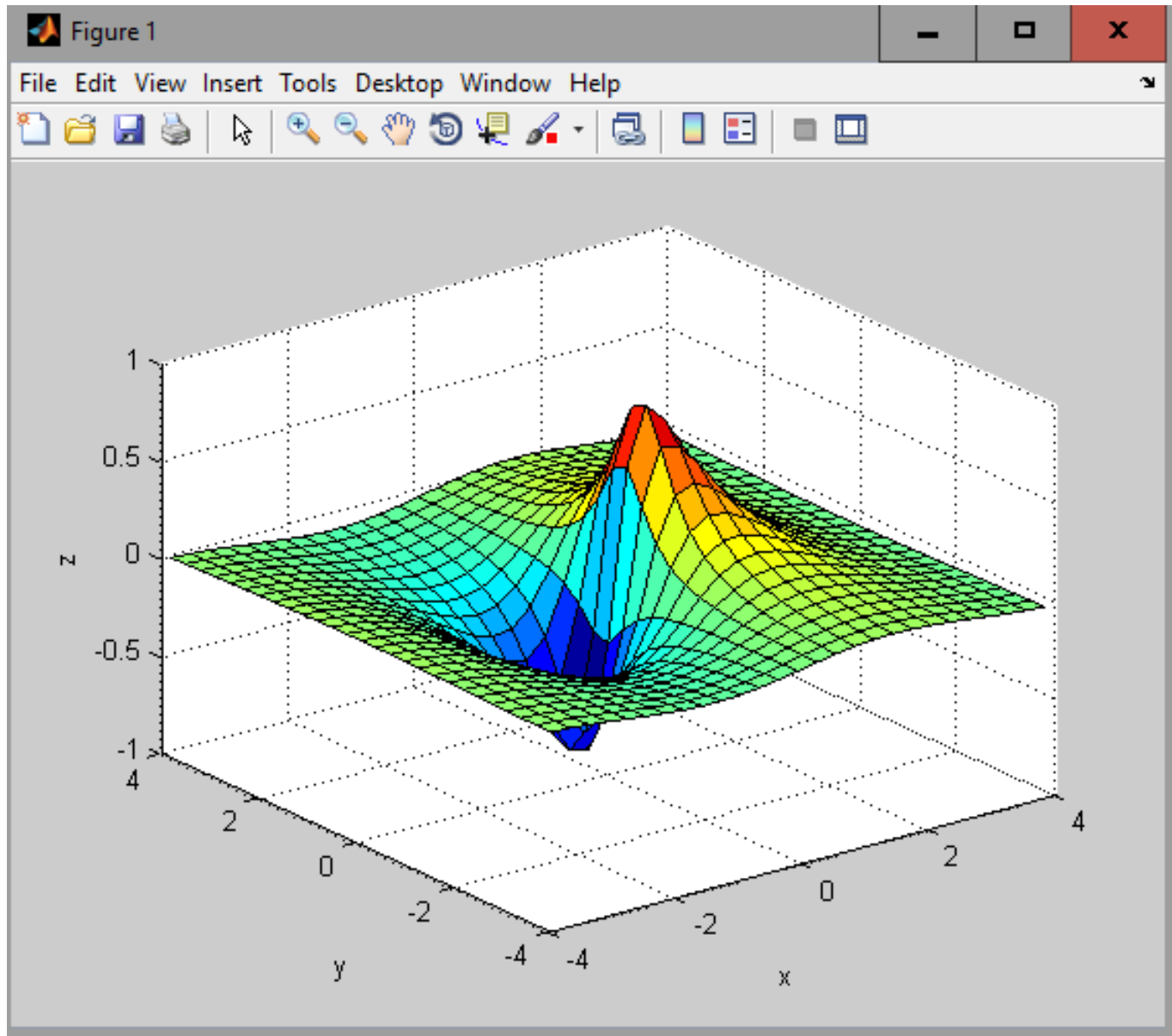
```
1 - [x,y]=meshgrid(-4:0.3:4);  
2 - z=sin(x) ./ (x.^2+y.^2+0.3);  
3 - surf(x,y,z);
```



# 3.4 Форматирование графиков



## 3.4 Форматирование графиков



# 3.5 Основы GUI

Видеоурок: <https://www.youtube.com/watch?v=1KKAlY3onI>

## Создание графического интерфейса пользователя (GUI) в MATLAB