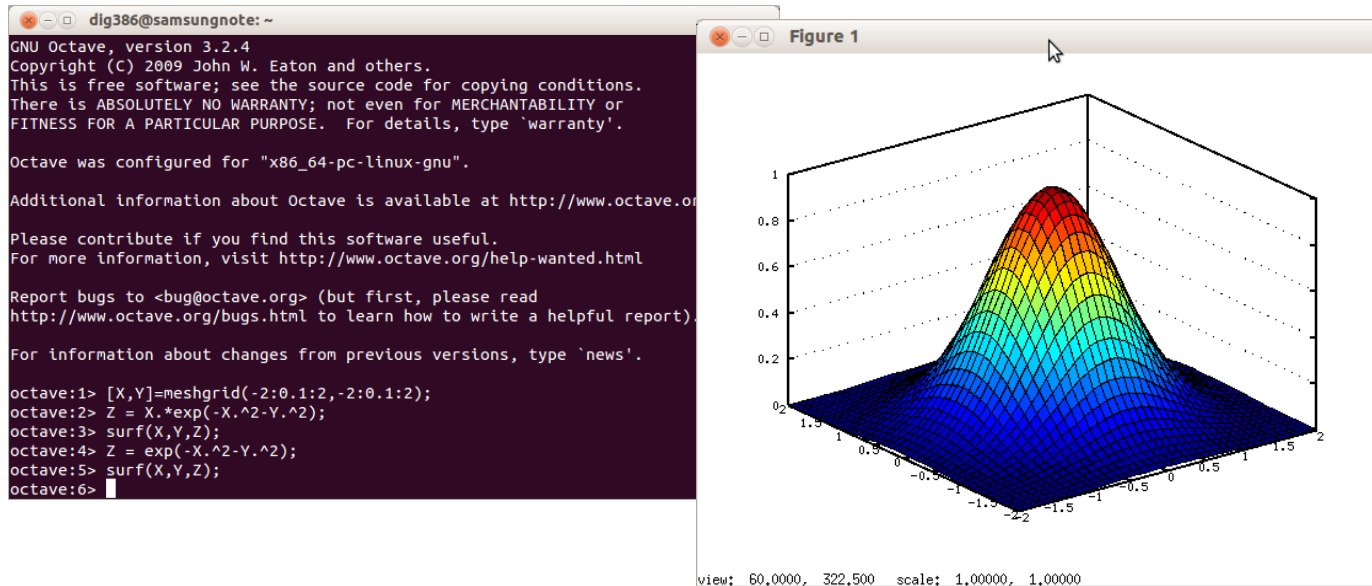


# Занятие 2. Основы работы в GNU Octave



## Краткое содержание занятия

1. Создание матриц и простейшие операции над ними
2. Решение систем уравнений.
3. Графики функций  $z=f(x)$  и  $z=f(x,y)$
4. Расчёт доверительных интервалов
5. Гистограммы

# Почему GNU Octave?

Готовые «мастера» в пакетах вроде Origin, Statgraphics и т.п.

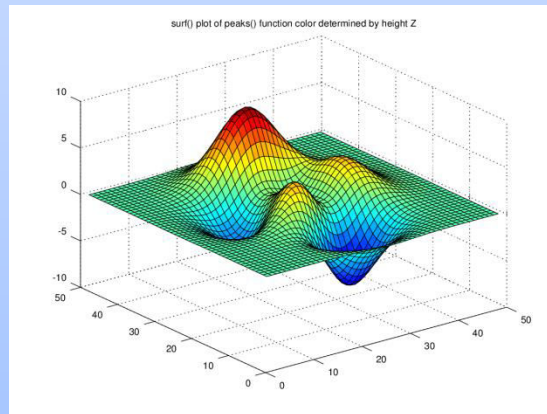
Просто и надёжно

НО! Теория оторвана от практики => сложно разобраться в принципах работы



Системы матричной алгебры: MATLAB, GNU Octave, SciLab

Разумный компромисс между сложностью C/Fortran и загадочностью «мастеров»



«Классические» языки C и Fortran

Даёт полное понимание происходящего и контроль над ним

НО! Трудоёмко, нужны навыки программиста



# Создание матриц

## Способ 1. Перечисление элементов в квадратных скобках

```
>> a=[1 2 3;4 5 6]
```

```
a =
```

```
1    2    3
4    5    6
```

```
>> b=[10 20 30;40 50 60];
```

```
>> disp(b)
```

```
10    20    30
40    50    60
```

**Примечание:** если в конце строки поставить точку с запятой, то матрица будет создана, но при этом не показана

## Способ 3. Оператор двоеточия

```
>> disp(10:15:70)
```

```
10    25    40    55    70
```

```
>> disp(1:5)
```

```
1    2    3    4    5
```

## Способ 2. Конкатенация существующих матриц

```
>> c = [a;b]; disp(c)
```

```
1    2    3
4    5    6
10   20   30
40   50   60
```

```
>> d = [a b]; disp(d)
```

```
1    2    3    10    20    30
4    5    6    40    50    60
```

## Способ 4. Специальные функции

```
>> x = eye(3);
```

```
>> disp(x)
```

```
Diagonal Matrix
```

```
1    0    0
0    1    0
0    0    1
```

См. также ones, zeros, repmat

# Создание матриц

## Способ 5. Выделение подматриц

```
>> x = magic(4)
    16     2     3    13
     5    11    10     8
     9     7     6    12
     4    14    15     1
```

```
>> disp(x(1:2,1:4))
```

```
    16     2     3    13
     5    11    10     8
```

```
>> disp(x(1:2:4,1:2:4))
```

```
    16     3
     9     6
```

```
>> disp(x(1,:))
```

```
    16     2     3    13
```

## Индексы начинаются с единицы!

Диапазоны индексов задаются с помощью конструкции с двоеточием.

: - все индексы по данной размерности

a:b – все индексы от a до b с шагом 1

a:s:b – все индексы от a до b с шагом s

## Способ 6. Работа с подматрицами

```
>> x = magic(4)
```

```
    16     2     3    13
     5    11    10     8
     9     7     6    12
     4    14    15     1
```

```
>> x(:,2)=-x(:,2); disp(x)
```

```
    16    -2     3    13
     5   -11    10     8
     9    -7     6    12
     4   -14    15     1
```

```
>> x([1 4],:)=x([4 1],:); disp(x)
```

```
     4   -14    15     1
     5   -11    10     8
     9    -7     6    12
    16    -2     3    13
```

```
>> x(2:4,2:4)=0;disp(x)
```

```
     4   -14    15     1
     5     0     0     0
     9     0     0     0
    16     0     0     0
```

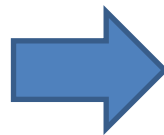
# Операции над матрицами

Octave	Линейная алгебра	Описание
$C=A.'$	$C = AT$	Транспонирование матрицы
$C=A'$	$C = A^*$	Эрмитово-сопряженная матрица
$C=A.^B$	$C_{ij} = A_{ij}^{B_{ij}}$	Поэлементное возведение в степень
$C=A*B$	$C = AB$	Умножение матриц
$C=A.*B$	$C_{ij} = A_{ij}B_{ij}$	Поэлементное умножение
$X=A \setminus B$	$AX = B \Leftrightarrow X = A^{-1}B$	Левое матричное деление
$X=A/B$	$XB = A \Leftrightarrow X = AB^{-1}$	Правое матричное деление
$X=A./B$	$X_{ij} = A_{ij}/B_{ij}$	Поэлементное деление
$C=A+B$	$C = A + B$	Сложение матриц
$C=A-B$	$C = A - B$	Вычитание матриц

**Различайте матричные и поэлементные операции!**

# Операции над матрицами

$$A = \begin{pmatrix} 1 & 2 \\ 3 & 4 \end{pmatrix}; B = \begin{pmatrix} 50 & 60 \\ 70 & 80 \end{pmatrix}$$



```
>> A = [1 2; 3 4];  
>> B = [50 60; 70 80];
```

## Операторы \* и .\*

**\***: умножение матриц

```
>> A*B  
    190    220  
    430    500
```

**.\***: умножение массивов

```
>> A.*B  
    50    120  
    210    320
```

## Операторы /, \ и ./

**\**: левое матричное деление  
(left matrix division)

```
>> A\B  
   -30   -40  
    40    50
```

$$A^{-1}B$$

**/**: правое матричное деление  
(right matrix division)

```
>> A/B  
    0.3000   -0.2000  
    0.2000   -0.1000
```

$$AB^{-1}$$

**./**: деление массивов

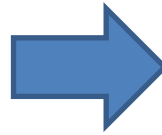
```
>> A./B  
    0.0200    0.0333  
    0.0429    0.0500
```

**Возможность скрытых ошибок!**

Если матрицы квадратные, то замена  
.\* на \* или / на ./ не приводит к  
останову программы!

# Операции над матрицами

$$A = \begin{pmatrix} 1 & 2 \\ 3 & 4 \end{pmatrix}; B = \begin{pmatrix} 50 & 60 \\ 70 & 80 \end{pmatrix}$$



```
>> A = [1 2; 3 4];  
>> B = [50 60; 70 80];
```

## Операторы ^ и .^

**^**: возведение матрицы в степень

```
>> A^2  
     7     10  
    15     22
```

**.^**: возведение массива в степень

```
>> A.^2  
     1     4  
     9    16
```

## Обратная матрица

(а) Функция **inv**

```
>> inv(A)  
    -2.0000    1.0000  
     1.5000   -0.5000
```

(б) Оператор **^**

```
>> A^(-1)  
    -2.0000    1.0000  
     1.5000   -0.5000
```

### Возможность скрытых ошибок!

Если матрицы квадратные, то замена .^ на ^ не приводит к останову программы!

# Решение системы линейных уравнений

## Исходная система уравнений

$$\begin{cases} x + y - z = 6 \\ 2x + 3y - 4z = 21 \\ 7x - y - 3z = 6 \end{cases}$$

## Шаг 1. Запись в матричном виде

$$A = \begin{pmatrix} 1 & 1 & -1 \\ 2 & 3 & -4 \\ 7 & -1 & -3 \end{pmatrix}; b = \begin{pmatrix} 6 \\ 21 \\ 6 \end{pmatrix}; Ax = b$$

## Шаг 2. Задать матрицы в Octave

```
>> A=[1 1 -1; 2 3 -4; 7 -1 -3];  
>> b=[6; 21; 6];
```

## Шаг 3. Решить систему

```
>> x=A\b  
x =  
 5.0753e-16  
 3.0000e+00  
-3.0000e+00
```

## Шаг 4. Проверить решение

```
>> disp(A*x);  
 6  
21  
 6
```

## Альтернативные способы

```
>> x=inv(A)*b;  
>> x=A^(-1)*b;
```



# Двухмерные графики

Построить график функции

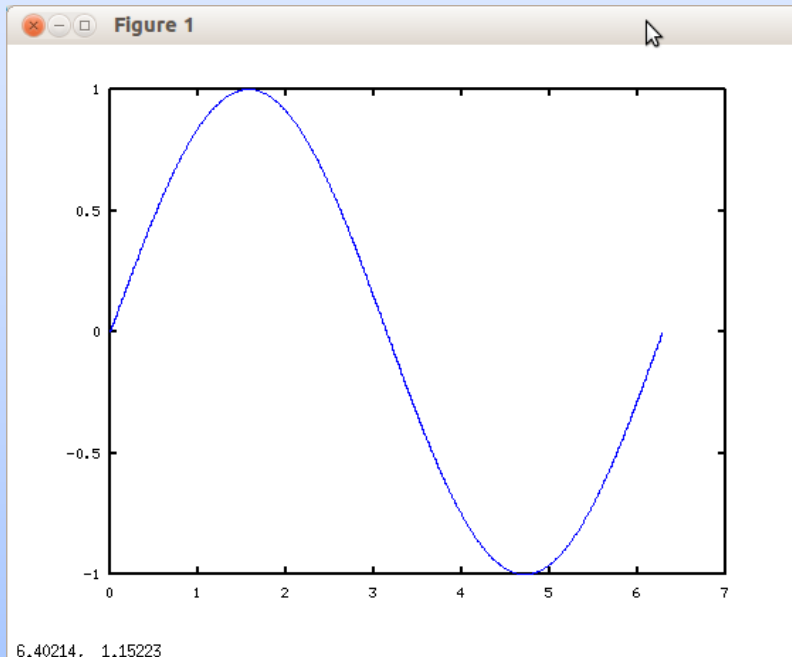
$$y = \sin(x); x \in [0, 2\pi]$$

Шаг 1. Рассчитать точки графика

```
>> x = 0:0.01:2*pi;  
>> y = sin(x);
```

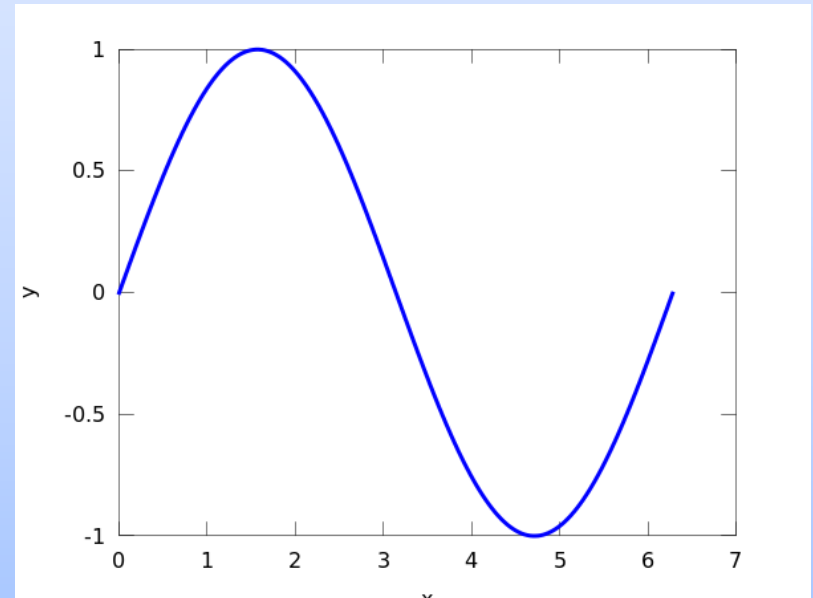
Шаг 2. Построить график

```
>> plot(x,y,'b-');
```



Шаг 3. Экспорт в файл

```
>> plot(x,y,'b-','LineWidth',3);  
>> xlabel('x'); ylabel('y');  
>> print('a.png','-dpng','-r75')
```



# Трёхмерные графики

**Построить график функции**

$$z = \exp(-x^2 - y^2);$$
$$x \in [-2, 2]; y \in [-2, 2]$$

```
>> % Рассчитать точки для графика  
>> [X,Y]=meshgrid(-2:0.1:2,-2:0.1:2)  
>> Z = exp(-X.^2-Y.^2)  
>> % Построить график  
>> surf(X,Y,Z);
```

## Работа функции meshgrid

```
>> [X,Y]=meshgrid(-3:1:3,-2:1:2)
```

Выход функции: матрицы X и Y

Диапазон и шаг по оси y

Диапазон и шаг по оси x

X =

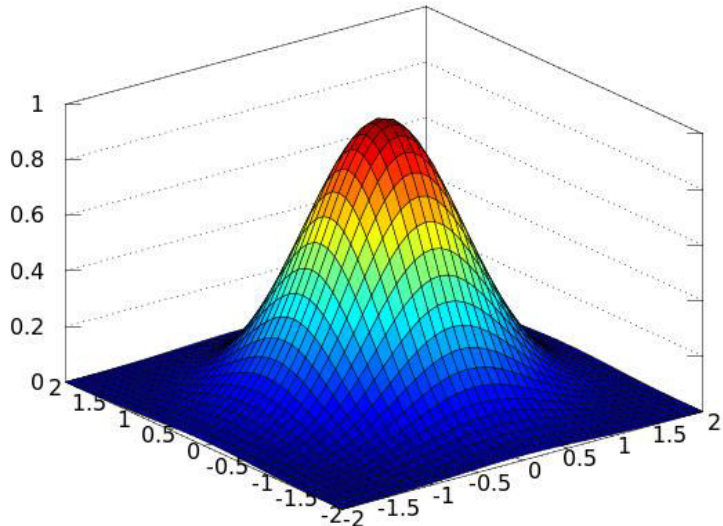
-3	-2	-1	0	1	2	3
-3	-2	-1	0	1	2	3
-3	-2	-1	0	1	2	3
-3	-2	-1	0	1	2	3
-3	-2	-1	0	1	2	3

Y =

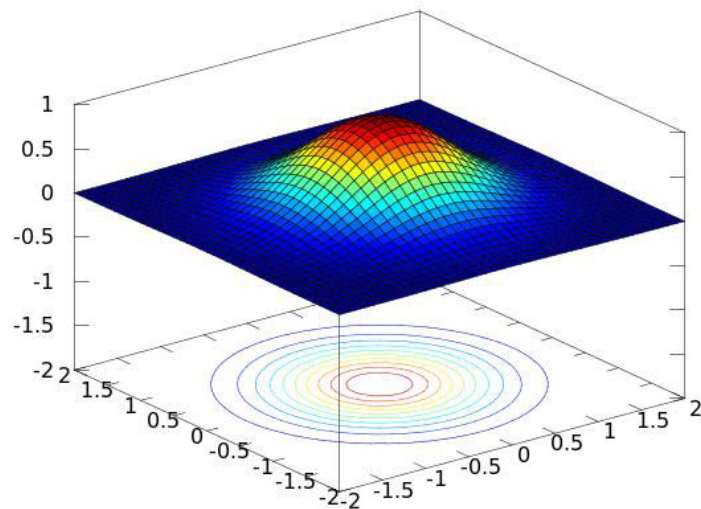
-2	-2	-2	-2	-2	-2	-2
-1	-1	-1	-1	-1	-1	-1
0	0	0	0	0	0	0
1	1	1	1	1	1	1
2	2	2	2	2	2	2

# Трёхмерные графики: вывод на экран

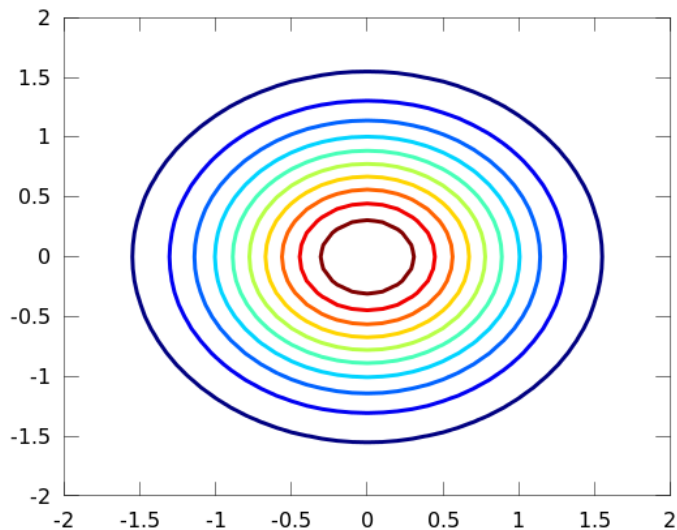
Функция surf: поверхность



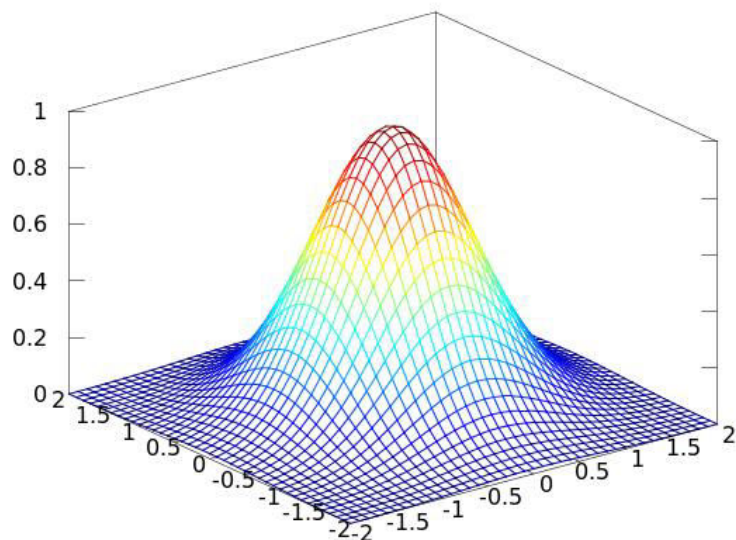
Функция surfc: поверхность + изолинии



Функция contour: только изолинии



Функция mesh: поверхность



# Среднее и доверительный интервал

Функция	Вызов	Описание
mean	mean(x)	Среднее значение вектора x
	mean(x,1)	Средние значения столбцов матрицы x
	mean(x,2)	Средние значения строк матрицы x
std	std(x)	Стандартное отклонение вектора x
	std(x,f)	$f = 0$ – несмещ.дисп., $f = 1$ – смещ.дисп.
	std(x,f,1)	Стандартное отклонение столбцов матрицы x
	std(x,f,2)	Стандартное отклонение строк матрицы x
tinv	tinv(p,f)	Левосторонний квантиль t-распределения
median	См. mean	Расчёт медианы
numel	numel(x)	Число элементов вектора или матрицы x
sum	См. mean	Расчёт суммы

```
>> x = [66.30 66.30 66.26 66.37 66.30];  
>> mean(x)  
ans = 66.306  
>> std(x)  
ans = 0.039749  
>> std(x)*tinv(0.975,numel(x)-1)/sqrt(numel(x))  
ans = 0.049355
```

# Функции распределения и квантили

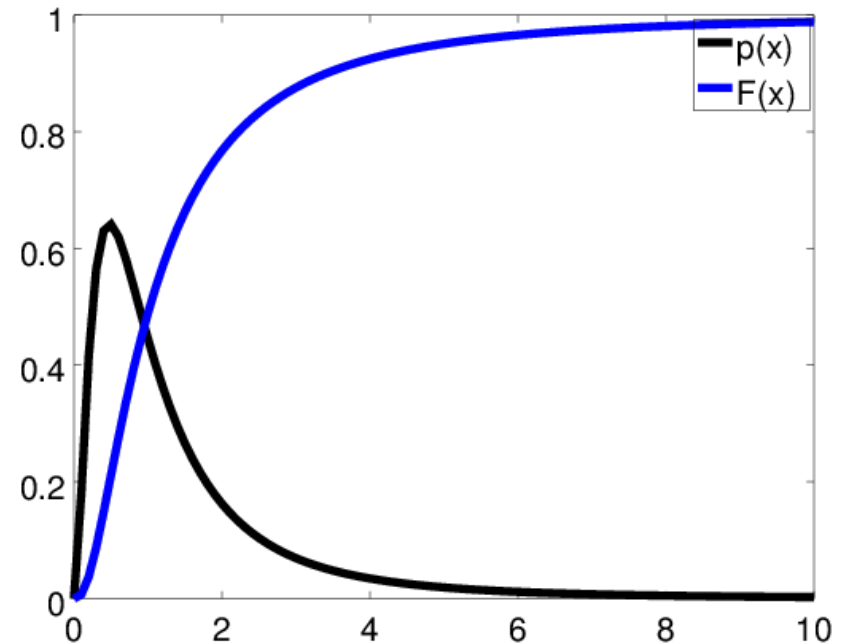
Распределение	$p(x)$	$F(x)$	Левосторонний квантиль
Нормальное	<code>normpdf (x , mu , s)</code>	<code>normcdf (x , mu , s)</code>	<code>norminv (p , mu , s)</code>
Стьюдента (t)	<code>tpdf (x , f)</code>	<code>tcdf (x , f)</code>	<code>tinv (p , f)</code>
Пирсона ( $\chi^2$ )	<code>chi2pdf (x , f)</code>	<code>chi2cdf (x , f)</code>	<code>chi2inv (p , f)</code>
Фишера (F)	<code>fpdf (x , f1 , f2)</code>	<code>fcdf (x , f1 , f2)</code>	<code>finv (p , f1 , f2)</code>

Pdf – probability density function (функция плотности вероятности)

Cdf – cumulative distribution function (интегральная функция распределения)

Inv – inverse function (обратная функция)

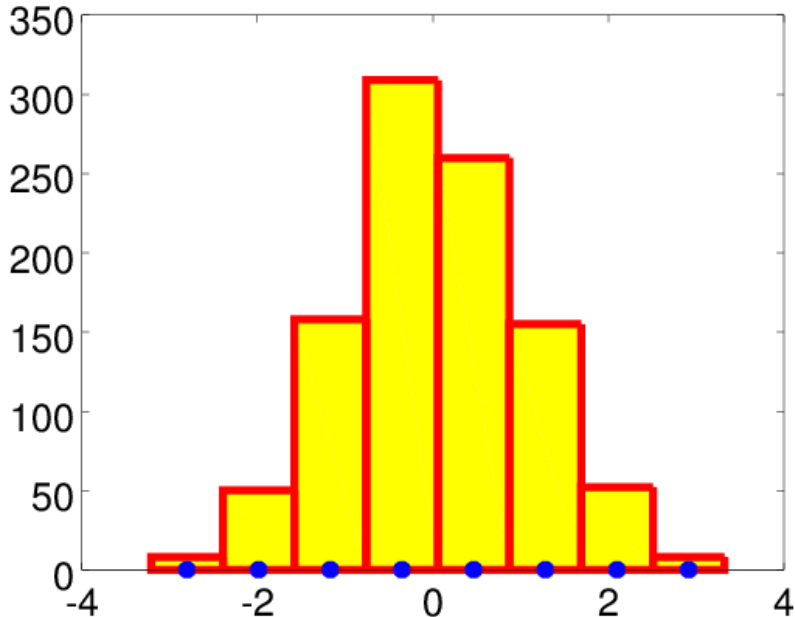
```
>> x = 0:0.1:10;
>> y1 = fpdf(x,6,5);
>> y2 = fcdf(x,6,5);
>> plot(x,y1,'k-',LineWidth,5);
>> hold on;
>> plot(x,y2,'b-',LineWidth,5);
>> hold off;
>> legend('p(x)', 'F(x)');
>> set(gca, 'FontSize', 20);
```



# Случайные числа и гистограммы

Функция	Вызов	Описание
<code>rand</code>	<code>rand(m,n)</code>	Матрица $m \times n$ из равномерно распределенных случайных чисел из интервала $[0;1]$
<code>randn</code>	<code>randn(m,n)</code>	Матрица $m \times n$ нормально распределенных случайных чисел ( $\mu = 0; \sigma = 1$ )
<code>hist</code>	<code>hist(x,n)</code> <code>[nn,xx]=hist(x,n)</code>	График с гистограммой Гистограмма в числах ( $x$ – данные, $n$ – число карманов, $nn$ – частоты, $xx$ – центры карманов)

```
>> x = randn(1,1000);  
>> hist(x,8, 'FaceColor',  
'yellow', EdgeColor, 'red',  
'LineWidth', 5);  
>> [nn,xx] = hist(x,8);  
>> hold on;  
>> plot(0,xx,'bo');  
>> hold off;
```



# Полезные команды

Команда	Значение
<code>cd имя_каталога</code>	Сменить текущий каталог
<code>clc</code>	Очистить консоль
<code>clear all</code>	Уничтожить все переменные (и ряд других структур вроде пользовательских типов данных, загруженных MEX-файлов и т.п.)
<code>close all</code>	Закрыть все окна с графиками
<code>dir (или ls)</code>	Вывести содержимое текущего каталога
<code>edit filename.m</code>	Отредактировать файл filename.m или создать его (если он отсутствует)
<code>figure</code>	Создать новое окно для вывода графиков
<code>help funcname</code>	Выдать справку по функции funcname
<code>hold on</code> <code>hold off</code>	Выводить новый график в существующее окно Выводить новый график в новое окно (старое закроется)
<code>pause</code>	Ждать нажатия ENTER пользователем
<code>pwd</code>	Узнать текущий каталог
<code>whos</code>	Вывести существующие переменные