

Matlab

- Variables are 2D arrays (matrices). Vectors/1D arrays are matrices/2D arrays with a single row or column
- `a=[1 2 3]` will assign vector/1D array [1 2 3] to variable `a` and print the result
- End command with a semicolon e.g. `a=[1 2 3];` to avoid printing result

Colon generates number sequence:

- `11:14` generates 11 12 13 14
- `-1:1` generates -1 0 1
- `3:0` generates Empty matrix

Specify step size with second colon:

- `1:3:12` generates 1 4 7 10
- `4:-1:1` generates 4 3 2 1
- `3:-0.5:2` generates 3.0 2.5 2.0

Variables

- `a=[1:3]` same as `a=[1 2 3]`
- Access element 3 of vector `a` using `a(3)` e.g. `a(3) = 4` produces `a=[1 2 4]`
- `end` is the highest index value e.g. `a(end)` is 4
- Using just `:` is equivalent to `1:end`

Select rows, columns of `a`:

- `a=[1 2 3];`
- `a(1,:)` is 1st row 1 2 3
- `a(:,1)` is 1st column 1

Access multiple elements of `a`:

- `a=[1 2 3];`
- `a(1:2)` is 1 2
- `a([1 3])` is 1 3

Variables

Use [] to build new arrays:

- `a=[1 2 3];`
- `[a 4 5]` is 1 2 3 4 5

Use “;” to add rows:

- `a=[1 2 3];`
- `[a; 4 5 6]` is matrix/2D

array:

1 2 3

4 5 6

Using conditions:

- `a=[1 2 3];`
- `a>2` is binary vector 0 0 1
- `find(a>2)` returns index 3

Operations

Operations + add, - subtract, * multiply, / divide work on vectors/matrices:

- `a=[1 2 3]; b=[4 5 6];`
- `a+b` is 5 7 9
- `a-b` is -3 -3 -3
- `a*2` is 2 4 6
- `a*b` gives an error since its a vector multiply
- Use “.” to make multiply and divide apply element by element i.e. `.*` and `./`
- `a.*b` is 4 10 18
- `a./b` is 0.25 0.40 0.50
- `a.^2` raises the elements of `a` to the power 2 giving 1 4 9

Operations

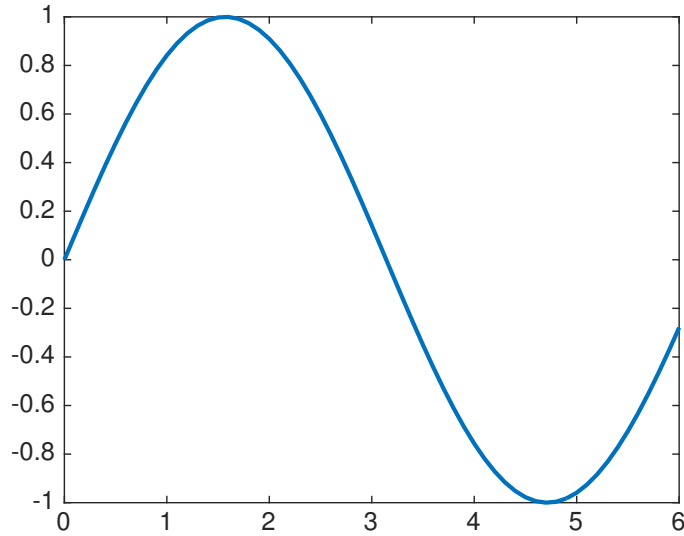
Functions `sin()`, `cos()`, `exp()` etc work on vectors/matrices:

- `a=[1 2 3];`
- `sin(a)` is 0.8415 0.9093 0.1411
- `exp(a)` is 2.7183 7.3891 20.0855

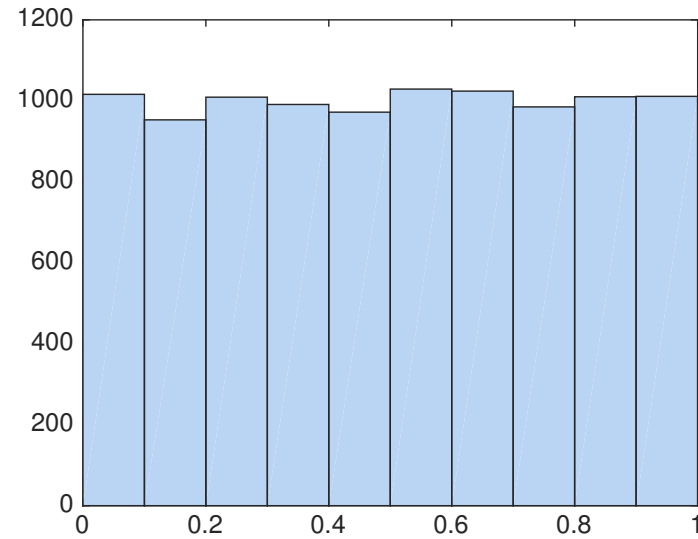
Use `rand()` to generate random numbers uniformly distributed between 0 and 1

- E.g `rand` might return 0.8147 on first call and 0.1270 on second call
- `rand(1,n)` returns a vector of n random numbers
- E.g. `rand(1,5)` might return 0.9134 0.6324 0.0975 0.2785
0.5469

Plotting



```
x=[0:0.1:6]; plot(x,sin(x))
```



```
x=rand(1,10000); hist(x)
```

- `hist(x)` calculates the number of times values occur in vector `x` and plots as histogram.
- Defaults to using 10 “bins”, but can change this to use `y` by `hist(x,y)`.

Useful Functions

- `help <command>`
- Comparisons: `==`, `=`, `>`, `<`, `>=`, `<=`
- Sizes of vectors/matrices: `length`, `size`
- `zeros`, `ones` all zeros and all ones vectors
- `xlim`, `ylim` plot axes ranges
- `xlabel`, `ylabel`, `title` plot labels
- `figure` open new figure
- `sum` sum up elements of a vector
- `mean`, `var` mean and variance of elements of a vector
- `find` find indices
- `nchoosek` return number of ways in which k objects can be drawn from a set of n objects
- `load`, `save` load and save data from file

Conditions and Loops

Conditions:

```
if <condition>
    <statement>
else
    <statement>
end
```

```
a=1; b=2
if a>b
    disp(a)
else
    disp(b)
end
```

Loops:

```
for <variable> = <vector>
    <statement>
end
```

```
for i=1:10
    disp(i)
end
```


Exercises

1. Generate $n = 10$ random numbers using `rand` function and calculate their sum
2. Repeat above to calculate random sum 100 times (it will change each time since its random) and plot a histogram of the values of the sum.
3. Now generate $n = 1000$ random numbers using `rand` function and calculate their sum. Repeat 100 times and plot a histogram of the values of the sum. How has it changed ?
4. Rather than plotting histogram of sums, plot histogram of sums divided by n (so divided by 10 in first case and by 1000 in second case). How do they differ ?

Exercises (extra)

1. Write a matlab program to enumerate all the different ways we can arrange the letters in the word "abc".