

Matlab Supplement

Machine Learning – CISC 5800
Dr Daniel Leeds

Programming in Matlab: Data types

- Numbers: -8.5, 0, 94
- Characters: 'j', '#', 'K' - always surrounded by single quotes
- Groups of numbers/characters – placed in between []
 - [5 10 12; 3 -4 12; -6 0 0] - spaces/commas separate columns, semi-colons separate rows
 - 'hi robot', ['h' 'i' ' ' 'robot'] - a collection of characters can be grouped inside a set of single quotes

2

Matrix indexing

- Start counting at 1
matrix1=[4 8 12; 6 3 0; -2 -7 -12];
matrix1(2,3) -> 0
 - Last row/column can also be designated by keyword “end”
matrix1(1,end) -> 12
 - Colon indicates counting up by increment
 - [2:10] -> [2 3 4 5 6 7 8 9 10]
 - [3:4:19] -> [3 7 11 15 19]
- matrix1(2,1:3) -> [6 3 0]

3

Vector/matrix functions

- vec1=[9, 3, 5, 7]; matrix2=[4.5 -3.2; 2.2 0; -4.4 -3];
- mean mean(vec1) -> 6
 - min min(vec1) -> 3
 - max max(vec1) -> ?
 - std std(vec1) -> 2.58
 - length length(vec1) -> ?
 - size size(matrix2) -> [3 2];

4

Extra syntax notes

- Semicolons suppress output of computations:

```
> a=4+5
a =
    9
> b=6+7;
>
```

- % starts a comment for the line (like // in C++)

- .*, ./, .^ performs element-wise arithmetic

```
>c=[2 3 4]./[2 1 2]
>c =
    [1 3 1]
>
```

5

Variables

- who, whos – list variables in environment

- Comparisons:

- Like C++: ==, <, >, <=, >=
- Not like C++: not ~, and &, or |

- Conditions:

- if(...), end;

- Loops:

- while(...), end;
- for x=a:b, end;

6

Data: .mat files

- **save** filename variableNames

- **load** filename

- Confirm correct directories:

- pwd – show directory (print working directory)
- cd – change directory
- ls – list files in directory

7

Define new functions: .m files

- Begin file with function header:

```
function output = function_name(input)
```

```
statement1;
```

```
statement2;
```

```
⋮
```

- Can allow multiple inputs/outputs

```
function [output1, output2] = function_name(input1, input2, input3)
```

8