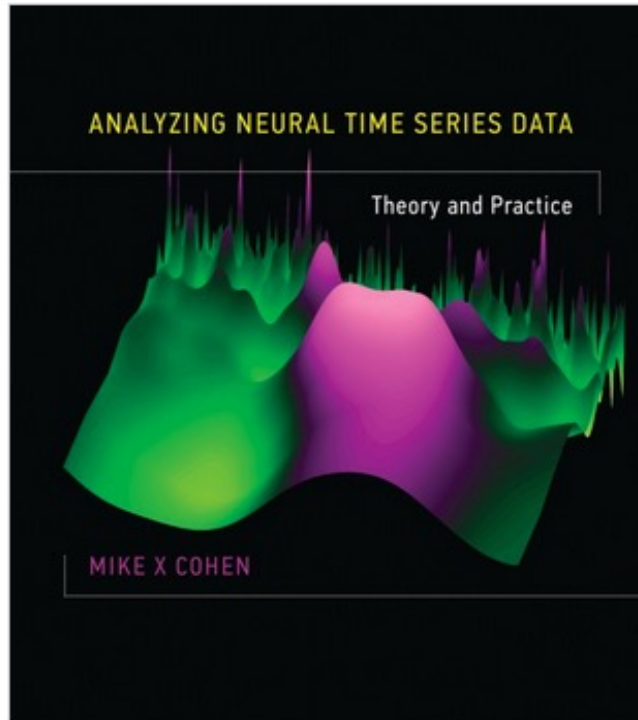


PART II:

- Working with time series data
- Oscillations and frequency decomposition
- Functional connectivity
- Population coding: working with high dimensional data
- Data mining approaches and best practices



Analyzing Neural Time Series Data

Theory and Practice

By Mike X Cohen

A comprehensive guide to the conceptual, mathematical, and implementational aspects of analyzing electrical brain signals, including data from MEG, EEG, and LFP recordings.

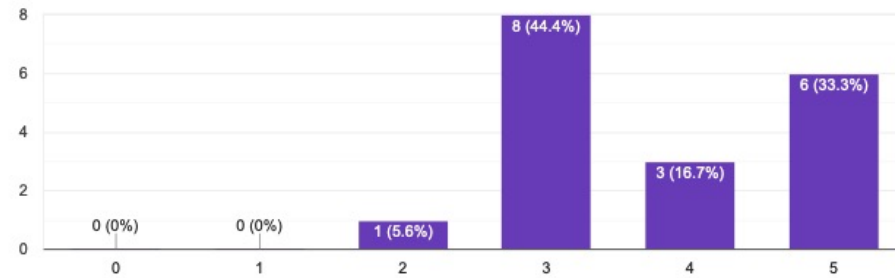
“In my experience teaching this material, most students look at an equation and slowly utter “okay”, as if they hope that by declaring the equation sensible, they will somehow understand what it means. But then they have a bewildered expression when it is time to turn to [...insert coding platform of choice...] and implement the equation.”

Why Matlab?

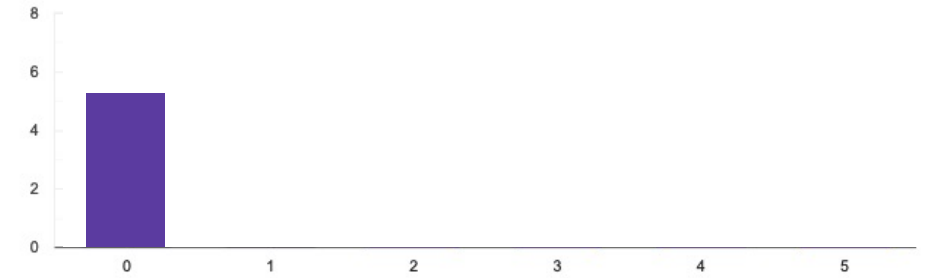
Why Matlab?

Familiarity with:

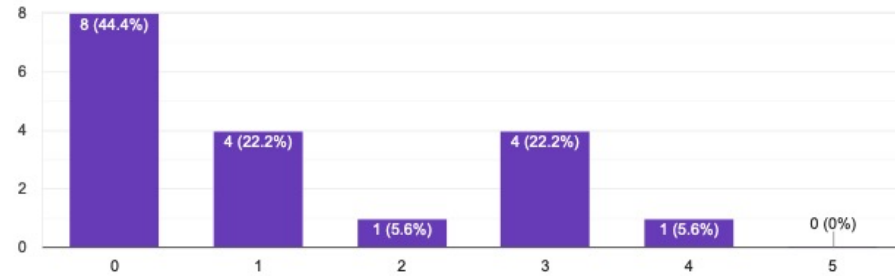
R



R



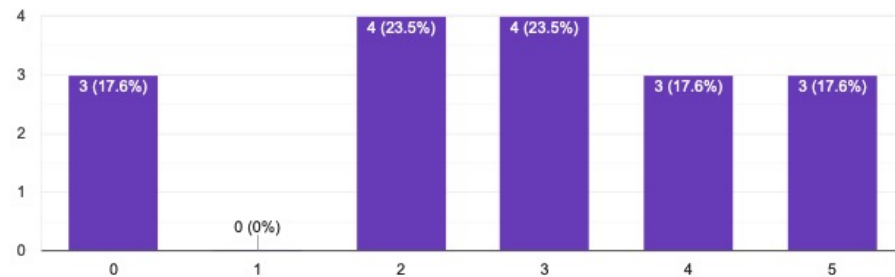
Matlab



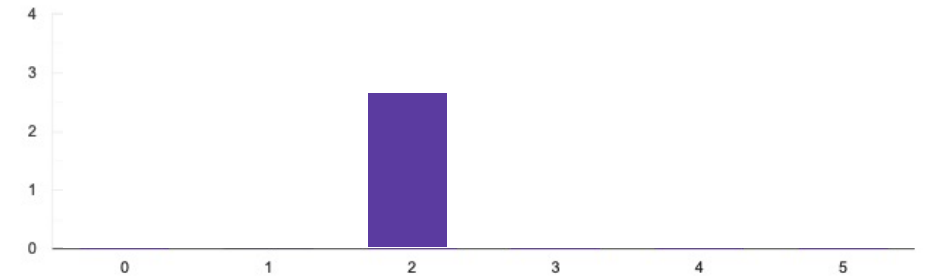
Matlab



Python



Python

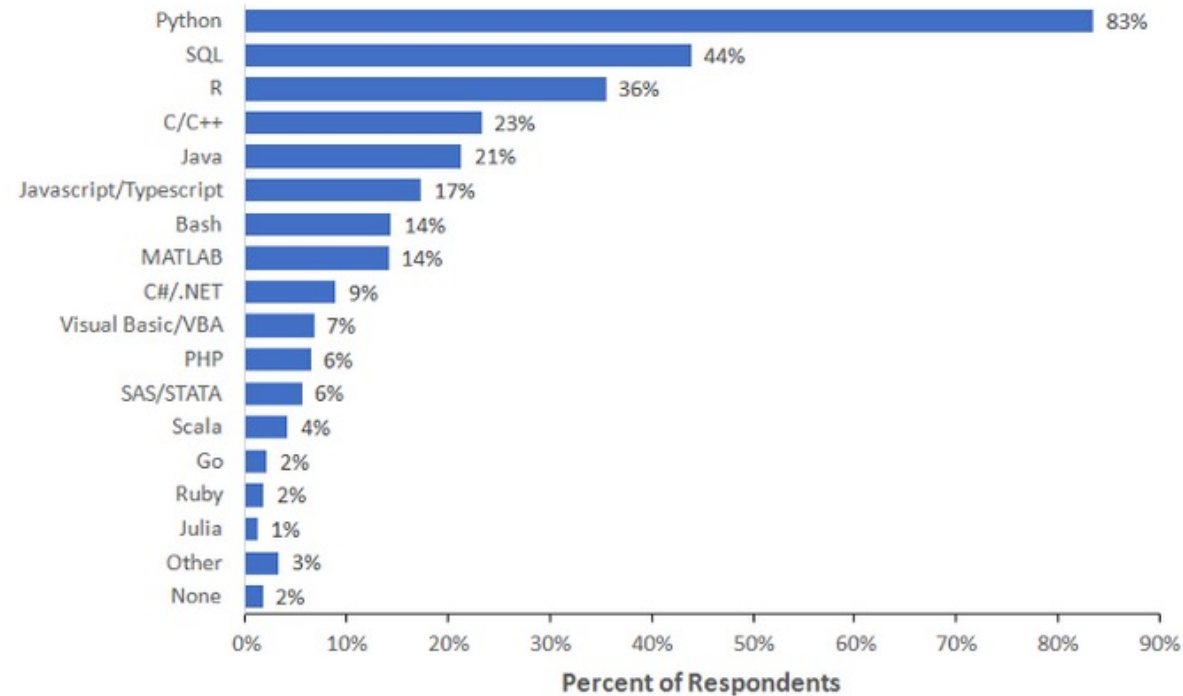


Never tried it → Proficient,
I use it regularly

Never tried it → Proficient,
I use it regularly

Programming Languages for Data Science

What programming language do you use on a regular basis?



Note: Data are from the 2018 Kaggle Machine Learning and Data Science Survey. You can learn more about the study here: <http://www.kaggle.com/kaggle/kaggle-survey-2018>. A total of 18827 respondents answered the question.

Vectors

$x1 =$

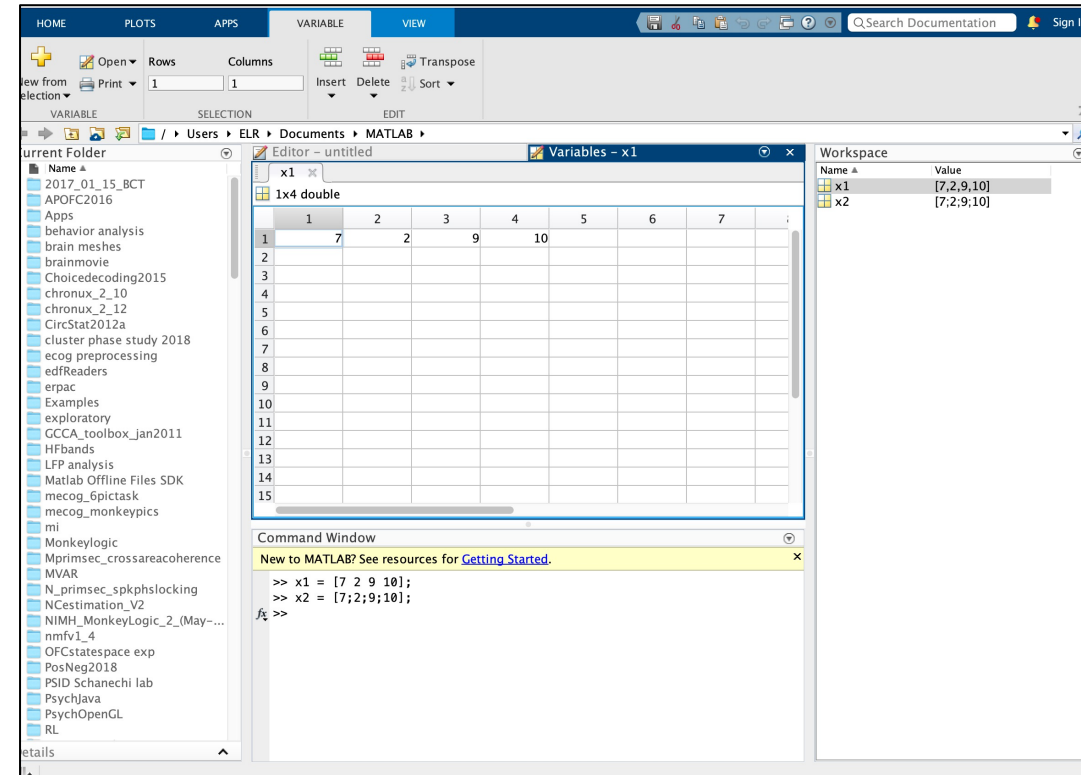
7	2	9	10
---	---	---	----

 Shape: 1 x 4
size(x1) = (1,4)

$x2 =$

7
2
9
10

 Shape: 4 x 1
Size(x2) = (4,1)



Vectors

$x1 =$

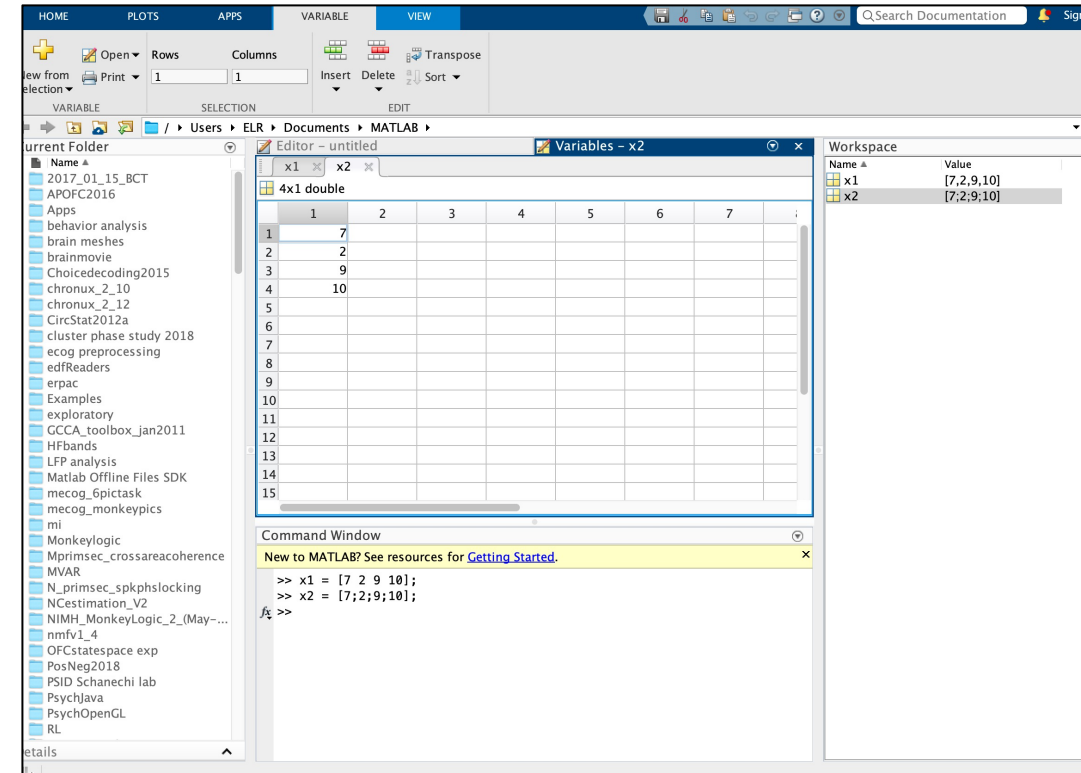
7	2	9	10
---	---	---	----

 Shape: 1 x 4
size(x1) = (1,4)

$x2 =$

7
2
9
10

 Shape: 4 x 1
Size(x2) = (4,1)



Vectors

$x1 =$

7	2	9	10
---	---	---	----

 Shape: 1 x 4
size(x1) = (1,4)

Indexing:
 $x1(1,3) = 9$
 $x2(3,1) = 9$

$x2 =$

7
2
9
10

 Shape: 4 x 1
Size(x2) = (4,1)

```
Command Window
New to MATLAB? See resources for Getting Started.
>> x2 = [7;2;9;10];
>> x1(1,3)

ans =

     9

>> x2(3,1)

ans =

     9

fx >>
```


Vectors

x1 =

7	2	9	10
---	---	---	----

 Shape: 1 x 4
size(x1) = (1,4)

x2 =

7
2
9
10

 Shape: 4 x 1
Size(x2) = (4,1)

Indexing:

x1(1,3) = 9

x2(3,1) = 9

****Note to python users****

Python starts indices at 0, not 1!

x1[1,3] does not exist!

x2[3,1] does not exist!

x1[0,2] = 9

x2[2,0] = 9

Vectors

$x1 =$

7	2	9	10
---	---	---	----

 Shape: 1 x 4
size($x1$) = (1,4)

$x2 =$

7
2
9
10

 Shape: 4 x 1
Size($x2$) = (4,1)

Indexing:
 $x1(1,3) = 9$
 $x2(3,1) = 9$

Transposing:
 $x1 = x2'$
 $x2 = x1'$

```
Command Window
New to MATLAB? See resources for Getting Started.

>> x1'
ans =
     7
     2
     9
    10

>> x2'
ans =
     7     2     9    10

fx >>
```

2D matrices

$y =$

5.2	3.0	4.5
9.1	0.1	0.3

Shape: 2 x 3
`size(y) = (2,3)`

The image shows the MATLAB R2021b interface. The main window displays the Command Window with the following code and output:

```
>> y = [5.2 3.0 4.5; 9.1 0.1 0.3];  
>> y  
  
y =  
  
    5.2000    3.0000    4.5000  
    9.1000    0.1000    0.3000  
  
>> y'  
  
ans =  
  
    5.2000    9.1000  
    3.0000    0.1000  
    4.5000    0.3000
```

The Workspace window on the right shows the variables defined:

Name	Value
ans	[5.2000, 9.1000; 3.0000, 0.1000]
x1	[7; 2; 9; 10]
x2	[7; 2; 9; 10]
y	[5.2000, 3.0000, 4.5000; 9.1000, 0.1000, 0.3000]

The Editor window shows the current folder structure, including folders like '2017_01_15_BCT', 'APOFC2016', 'Apps', 'behavior analysis', 'brain meshes', 'brainmovie', 'Choicedecoding2015', 'chronux_2_10', 'chronux_2_12', 'CircStat2012a', 'cluster phase study 2018', 'ecog preprocessing', 'edfReaders', 'erpac', 'Examples', 'exploratory', 'GCCA_toolbox_jan2011', 'HFbands', 'LFP analysis', 'Matlab Offline Files SDK', 'mecog_6pictask', 'mecog_monkeypics', 'mi', 'Monkeylogic', 'Mprimsec_crossareacoherence', 'MVAR', 'N_primsec_spkphslocking', 'NCestimation_V2', 'NIMH_MonkeyLogic_2_(May-...', 'nmfv1_4', 'OFCstatespace exp', 'PosNeg2018', 'PSID Schanechi lab', 'Psychjava', 'PsychOpenGL', and 'RL'.

2D matrices

y =

5.2	3.0	4.5
9.1	0.1	0.3

Shape: 2 x 3
size(y) = (2,3)

Indexing:

y(1,2) = 3.0

y(2,3) = 0.3

In python:

y(1,2) = 0.3

y(2,3) does not exist

Transposing:

```
Command Window
New to MATLAB? See resources for Getting Started.

>> y = [5.2 3.0 4.5;9.1 0.1 0.3];
>> y

y =

    5.2000    3.0000    4.5000
    9.1000    0.1000    0.3000

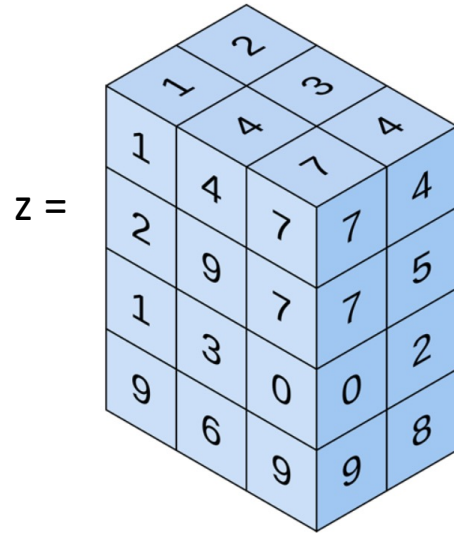
>> y'

ans =

    5.2000    9.1000
    3.0000    0.1000
    4.5000    0.3000

fx >>
```

3D matrices



Shape: 4 x 3 x 2
size(z) = (4,3,2)

MATLAB R2021b - academic use

HOME PLOTS APPS VARIABLE VIEW

Open Rows Columns Insert Delete Transpose
New from Selection Print Sort

VARIABLE SELECTION EDIT

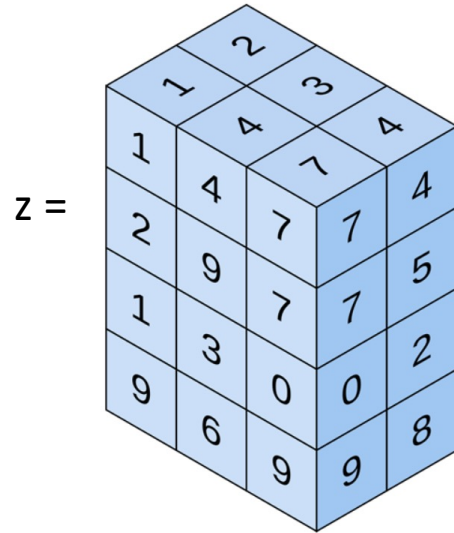
Current Folder
Name
2017_01_15_BCT
APOFC2016
Apps
behavior analysis
brain meshes
brainmovie
ChoiDecoding2015
chronux_2_10
chronux_2_12
CircStat2012a
cluster phase study 2018
ecog_preprocessing
edfReaders
erpac
Examples
exploratory
GCCA_toolbox_jan2011
HFbands
LFP analysis
Matlab Offline Files SDK
mecog_6pictask
mecog_monkeypicks
mi
Monkeylogic
Mprimsec_crossareacoherence
MVAR
N_primsec_spkphslocking
NCestimation_V2
NIMH_MonkeyLogic_2_(May-...
nmfv1_4
OFCstatespace exp
PosNeg2018
PSID_Schanechi lab
PsychJava
PsychOpenGL
RL

Editor - untitled
z 4x3x2 double
val(:,:,1) =
1 4 7
2 9 7
1 3 0
9 6 9
val(:,:,2) =
2 3 4
1 4 5
6 8 2
9 3 8

Command Window
New to MATLAB? See resources for [Getting Started.](#)
z(:,:,1) =
1 4 7
2 9 7
1 3 0
9 6 9
z(:,:,2) =
2 3 4
1 4 5
6 8 2
9 3 8
f3 >>

Workspace
Name Value
ans [5.2000,9.1000;3...
x1 [7,2,9,10]
x2 [7,2,9,10]
z 4x3x2 double

3D matrices



Shape: 4 x 3 x 2
size(z) = (4,3,2)

Indexing:

$z(2,2,1) = 9$

$z(1,3,2) = 4$

Transposing: Not defined

Matrix Algebra

just a few things to keep in mind...

1. For element-by-element addition/subtraction your dimensions must match

k =

4
12
1
23
9
18

m =

34
8
1
13

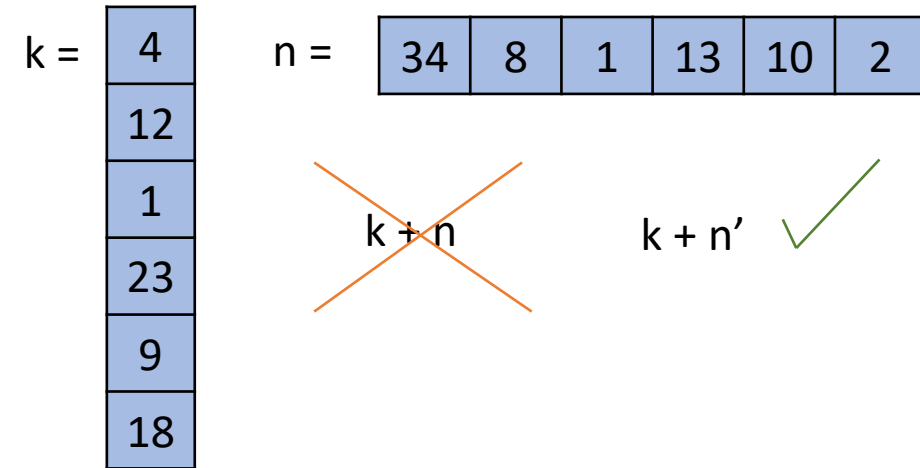
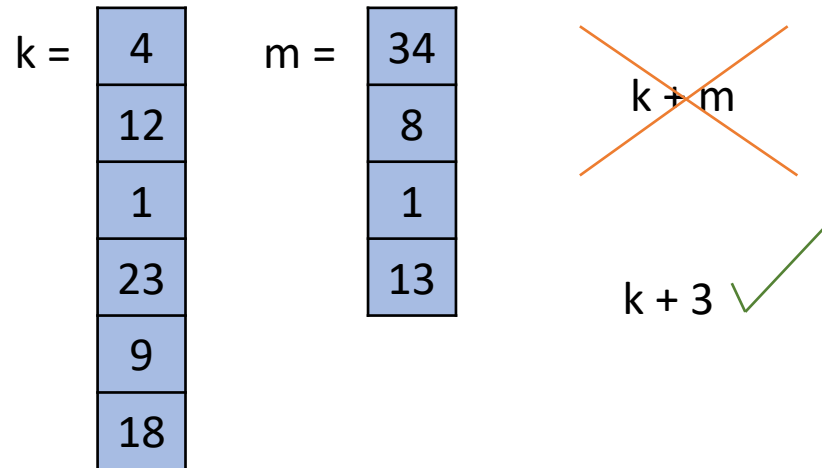
~~k + m~~

k + 3 ✓

Matrix Algebra

just a few things to keep in mind...

1. For element-by-element addition/subtraction your dimensions must match



Matrix Algebra

just a few things to keep in mind...

1. For element-by-element addition/subtraction your dimensions must match
2. Multiplication/division are assumed to be matrix operations

k =

4
12
1
23
9
18

m =

34
8
1
13

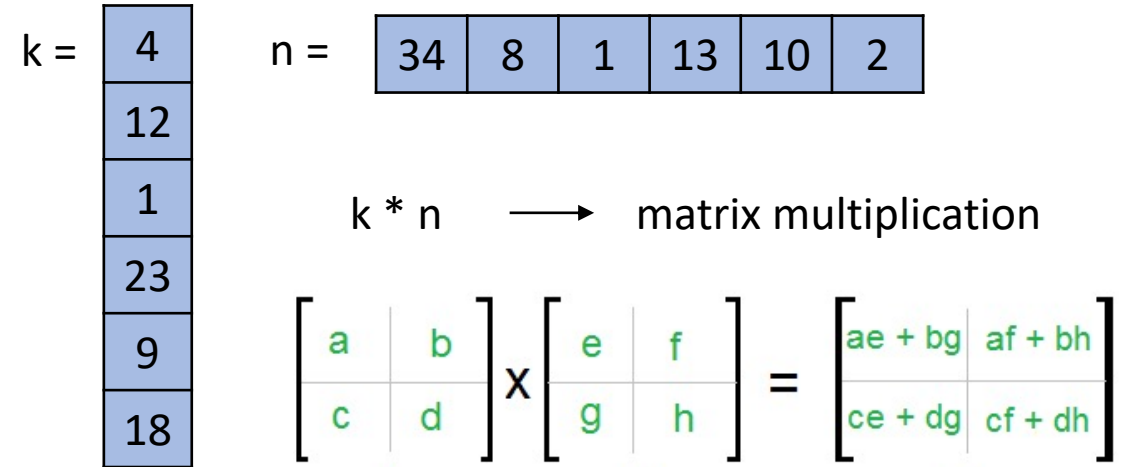
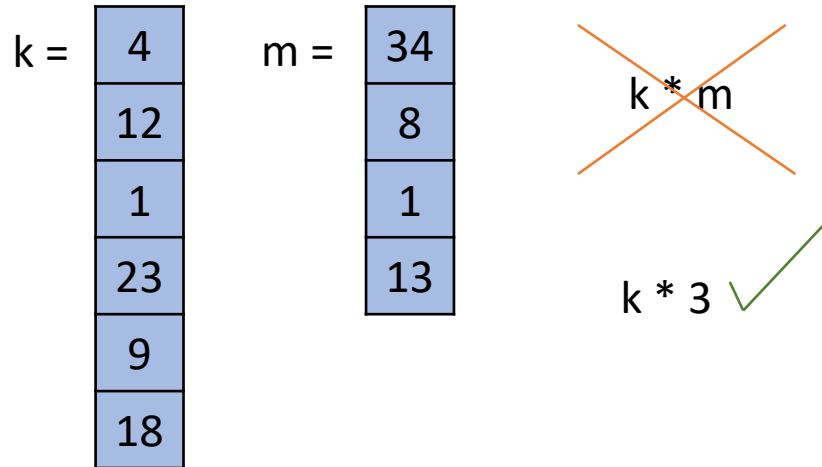
~~k * m~~

k * 3 ✓

Matrix Algebra

just a few things to keep in mind...

1. For element-by-element addition/subtraction your dimensions must match
2. Multiplication/division are assumed to be matrix operations



Matrix Algebra

just a few things to keep in mind...

1. For element-by-element addition/subtraction your dimensions must match
2. Multiplication/division are assumed to be matrix operations
-> use `.*` or `./` for element-by-element operations

k =

4
12
1
23
9
18

m =

34
8
1
13

~~k * m~~

k * 3 ✓

~~k .* m~~

k =

4
12
1
23
9
18

n =

34	8	1	13	10	2
----	---	---	----	----	---

k * n → matrix multiplication

~~k * n'~~

k .* n' ✓

Matrix Algebra

just a few things to keep in mind...

1. For element-by-element addition/subtraction your dimensions must match
2. Multiplication/division are assumed to be matrix operations
-> use `.*` or `./` for element-by-element operations

k =

4	7
12	3
1	10
23	11
9	0
18	2

m =

34
8
1
13
10
2

n =

34	8	1	13	10	2
5	9	22	6	14	1

k - m ✓
~~k - n~~
~~k * m~~
k * n → matrix multiplication
~~k * n'~~
k .* n' ✓
k ./ n' ✓
~~k .* m'~~

“for” loops

k =	4	7	n =	34	8	1	13	10	2
	12	3		5	9	22	6	14	1
	1	10							
	23	11							
	9	0							
	18	2							

for each row of k: find the row mean, multiply by the corresponding column of the first row of n, then subtract the corresponding column of the second row of n

make a new variable (e.g. 'i')

define how 'i' will change with each loop

for i = 1 : length(k(:,1))

Y(i,1) = (mean(k(i,:)) * n(1,i)) - n(2,i)

end

Same answer with matrix operations

$Y = (\text{mean}(k,2) \cdot n(1,:)') - n(2,:)$

*starting with 'for' indicates
you're opening a for loop*

for loop in R

```
Console Terminal x
~/
> for(i in 1:5) { # Basic for-loop
+   print(paste("This is step", i))
+ }
[1] "This is step 1"
[1] "This is step 2"
[1] "This is step 3"
[1] "This is step 4"
[1] "This is step 5"
> |
```

for loop in python

```
PyScripter - C:\Users\sah\Documents\Python Programs\testarea
File Edit Search View Project Run Tools Help
testarea.py x
Python Interpreter
sum = 0
for i in range(10):
    sum = sum + i
    print sum
6
10
15
21
28
36
45
>>>
```

Self-guided exercises on matrix operations and for loops