

Exercise: CasADi basics

For these exercises, assume the following objects are defined:

```
import casadi.*
x = MX.sym('x');      y = MX.sym('y')      % scalar symbol
p = MX.sym('p',2);    A = MX.sym('A',4,5)  % vector and matrix symbol
f = Function('f',{x,y},{sin(x)*y});
g = Function('g',{x},{sqrt(x),x^2});
h = Function('h',{x},{sin(x)*y});
F = @(x,y) sin(x)*y;
```

1 What's the type?

For each code snippet, encircle the character corresponding to the most appropriate data-type. When in doubt, enter the code snippet in Matlab and use `class(.)`.

Nr	Code	MX (constant)	MX (other)	DM	Function	non-CasADi type
1	5	v	r	g	y	e
2	5+x	r	l	g	x	p
3	x-x	e	y	d	m	g
4	x==x	c	z	h	y	v
5	x==y	k	t	e	j	a
6	A*x	b	r	d	e	y
7	A*0	o	k	n	y	d
8	gradient(sin(x),x)	f	e	h	r	g
9	gradient(x,x)	n	k	e	g	r
10	p(2)	q	c	a	g	i
11	A(:,1)	s	e	l	k	d
12	f(1,2)	o	c	p	m	x
13	f(1,y)	o	h	g	y	m
14	g(9)	n	p	a	v	f
15	g(x)	x	l	p	f	v
16	h(x)	t	o	g	r	w
17	f.expand()	p	w	a	g	v
18	F	u	d	h	q	r
19	F(x,y)	s	a	l	v	b
20	F(1,2)	p	q	x	y	m

Solution: electro-encephalogram

2 Correct the error

Goal 1: Compose a Function $x \mapsto \sin(x)$

Tried: `Function('m',[x],[sin(x)])`

No matching function for overload function 'new_Function'.

Possible prototypes are:

`FUNCTION(char,{MX},{MX})`

You have: char, MX, MX

Correction:

Goal 2: Get first element of p vector

Tried: `p[1]`

Invalid expression. When calling a function or indexing a variable, use parentheses. Otherwise, check for mismatched delimiters.

Correction:

Goal 3: Get last element of p vector

Tried: `p(3)`

Out of bounds error. Got elements in range [3,3], which is outside the range [-1,3).

Correction:

Goal 4: Concatenate two symbolic expressions

Tried: `M=zeros(2,1);M(1)=x;M(2)=y;`

Conversion to double from casadi.MX

Correction:

Goal 5: Call Function f numerically for $x=1,y=2$

Tried: `f{1,2}`

Brace indexing is not supported for variables of this type.

Correction:

Goal 6: Call Function f numerically with elements of p

Tried: `f(p)`

Incorrect number of inputs: Expected 2, got 1

Correction:

Goal 7: Create a Function that returns the square of f 's output

Tried: `f^2`

Undefined operator '^' for input arguments of type 'casadi.Function'.

Correction:

Goal 8: Evaluate h for $x = 5$

Tried: `h(5)`

Cannot evaluate "f:(i0)->(o0) MXFunction" since variables [y] are free

Correction:

Goal 9: Compose a Function $x \mapsto 5$

Tried: `Function('m',{5},{x})`

Xfunction input arguments must be purely symbolic.

Argument 0(i0) is not symbolic.

Correction:

Goal 10: Compose a Function $x \mapsto g(x)$

Tried: `Function('m',{x},{g})`

No matching function for overload function 'new_Function'.

Possible prototypes are:

`FUNCTION(char,{MX},{MX})`

You have: `char, {MX}, {Function}`

Correction:

Goal 11: Compose a Function with two input arguments where the second one is ignored

Tried: `Function('m',{x,x},{x})`

The input expressions are not independent:

0: `x`

1: `x.`

Correction:

Goal 12: Compose a Function $x \mapsto \sin(x)$

Tried: `Function('m',{sin(x)},{x})`

Xfunction input arguments must be purely symbolic.

Argument 0(i0) is not symbolic

Correction:

Solutions:

1. `Function('m',{x},{sin(x)})`
2. `p(1)`
3. `p(2)`
4. `M=MX(zeros(2,1));M(1)=x;M(2)=y;`
or `M=MX.zeros(2,1);M(1)=x;M(2)=y;`
or `M=[x;y];` (preferred)
5. `f(1,2)`
6. `f(p(1),p(2))`
7. `Function('m',{x,y},{f(x,y)^2})`
8. `h(MX(5))`
9. `Function('m',{x},{5})`
10. `Function('m',{x},{g(x)})`
11. `Function('m',{x,MX()}},{x})`
12. `Function('m',{x},{sin(x)})`

3 What's the type (part 2)?

CasADi Functions can also be called with keyword-value arguments...

```
f2 = Function('f2',{x,y},{sin(x)+y},{'x','y'},{'z'});
g2 = Function('g2',{x},{sqrt(x),x^2},{'x'},{'p0','p1'});
```

Nr	Code	MX (const.)	MX (other)	DM	Func.	non-CasADi	Error
1	<code>f2(3)</code>	a	s	r	e	m	c
2	<code>f2(3,4)</code>	b	q	a	r	w	u
3	<code>f2('y',4,'x',3)</code>	a	y	p	b	r	g
4	<code>getfield(f2('y',4,'x',3),'z')</code>	x	r	d	j	g	w
5	<code>getfield(f2('x',x,'y',4),'z')</code>	l	i	k	c	v	y
6	<code>f('x',3)</code>	f	a	p	g	y	o
7	<code>f(3)</code>	x	a	c	k	l	v
8	<code>f2('x',3)</code>	v	p	s	f	a	y
9	<code>f2()</code>	i	b	y	j	s	z
10	<code>getfield(f2(),'z')</code>	j	y	c	n	b	w
11	<code>g2(x)</code>	p	u	i	b	c	h
12	<code>g2('x',3)</code>	g	m	b	o	l	p
13	<code>g(3)</code>	y	o	a	g	n	u
14	<code>g2([])</code>	m	l	r	y	n	d

Solution: cardiovascular