FIR filter design with Python and SciPy

Matti Pastell
http://mpastell.com

15th April 2013

1 Introduction

This an example of a document that can be published using Pweave. Text is written using LaTeX and code between <<>> and @ is executed and results are included in the resulting document.

You can define various options for code chunks to control code execution and formatting (see Pweave docs).

2 FIR Filter Design

We'll implement lowpass, highpass and ' bandpass FIR filters. If you want to read more about DSP I highly recommend The Scientist and Engineer's Guide to Digital Signal Processing which is freely available online.

2.1 Functions for frequency, phase, impulse and step response

Let's first define functions to plot filter properties.

	salary_grade	filled_positions	num_positions	vacancies	salary_per_gdp_high	salary_per_gdp_low	Cadre	filled_in_kampala	filled_not_in_kampala_not_htr	filled_in_htr
0	NaN	110	157	47	NaN	NaN	Unknown cadre	NaN	NaN	NaN
1	U1SE	4	51	47	19.242749	14.670851	Consultant	3.0	1.0	0.0
2	U2(SC)	4	40	36	11.490033	9.647355	Medical Officer Special Grade	3.0	1.0	0.0
3	U3(SC)	4	19	15	7.761599	6.722766	Principle AO	0.0	3.0	1.0
4	U4(SC)	41	169	128	6.574276	6.082162	Senior AO	16.0	22.0	3.0
5	U5(SC)	181	599	196	4.426169	3.489347	AO	10.0	148.0	23.0
6	U7U	61	472	263	2.108906	1.766216	Anesthetic Assistant	0.0	53.0	6.0
7	U8U	35	230	119	1.323406	1.171507	Anesthetic Attendent	5.0	21.0	9.0

Table 1: Table to test captions and labels

m = [1, 1; 2, 4; 3, 9]; filename = ” t = matrix2latex(m, filename)

from matrix2latex import matrix2latex import numpy data = numpy.array(r) matrix2latex(data, ’outputfile.tex’, ’table’, ’tabular’, ’small’)

2.2 Lowpass FIR filter

Designing a lowpass FIR filter is very simple to do with SciPy, all you need to do is to define the window length, cut off frequency and the window.

The Hamming window is defined as: $w(n) = \alpha - \beta \cos \frac{2\pi n}{N-1}$, where $\alpha = 0.54$ and $\beta = 0.46$

The next code chunk is executed in term mode, see the source document for syntax. Notice also that Pweave can now catch multiple figures/code chunk.

```
n = 61
a = signal.firwin(n, cutoff = 0.3, window = "hamming")
```

```
------------------------------------------------------------NameError
Traceback (most recent call last)<ipython-input-1-0edf555780c5> in
<module>()
----> 1 a = signal.firwin(n, cutoff = 0.3, window = "hamming")
NameError: name 'signal' is not defined
```

```
#Frequency and phase response
mfreqz(a)
```

```
------------------------------------------------------------NameError
Traceback (most recent call last)<ipython-input-1-4b99431c27e4> in
<module>()
----> 1 mfreqz(a)
NameError: name 'mfreqz' is not defined
```

```
show()
```

```
------------------------------------------------------------NameError
Traceback (most recent call last)<ipython-input-1-9eb7fa60ac78> in
<module>()
----> 1 show()
NameError: name 'show' is not defined
```

```
#Impulse and step response
figure(2)
```

```
------------------------------------------------------------NameError
Traceback (most recent call last)<ipython-input-1-7bb9eaa62d9f> in
<module>()
----> 1 figure(2)
NameError: name 'figure' is not defined
```

```
impz(a)
```

```
---------------------------------------------------------------NameError
Traceback (most recent call last)<ipython-input-1-e3d06ad960d4> in
<module>()
----> 1 impz(a)
NameError: name 'impz' is not defined

show()

---------------------------------------------------------------NameError
Traceback (most recent call last)<ipython-input-1-9eb7fa60ac78> in
<module>()
----> 1 show()
NameError: name 'show' is not defined
```

2.3 Highpass FIR Filter

Let's define a highpass FIR filter:

```
n = 101
a = signal.firwin(n, cutoff = 0.3, window = "hanning", pass_zero=False)
mfreqz(a)
show()

---------------------------------------------------------------NameError
Traceback (most recent call last)<ipython-input-1-f567ff2f04e1> in
<module>()
      1 n = 101
----> 2 a = signal.firwin(n, cutoff = 0.3, window = "hanning",
pass_zero=False)
      3 mfreqz(a)
      4 show()
NameError: name 'signal' is not defined
```

2.4 Bandpass FIR filter

Notice that the plot has a caption defined in code chunk options.

```
n = 1001
a = signal.firwin(n, cutoff = [0.2, 0.5], window = 'blackmanharris', pass_zero = F
mfreqz(a)
show()

---------------------------------------------------------------NameError
Traceback (most recent call last)<ipython-input-1-360bebdd9724> in
<module>()
      1 n = 1001
----> 2 a = signal.firwin(n, cutoff = [0.2, 0.5], window =
'blackmanharris', pass_zero = False)
      3 mfreqz(a)
      4 show()
NameError: name 'signal' is not defined
```

Figure 1: Bandpass FIR filter.