Math 320 Homework 4.8

Chris Rytting

November 20, 2015

4.37

We know that

$$\psi(x) = \varphi(2x) - \varphi(2x-1)$$

and that

$$\psi_{jk}(x) = \psi(2^j x - k)$$

yielding

$$\begin{aligned} \psi_{jk}(x) &= \varphi(2(2^j x - k)) - \varphi(2(2^j x - k) - 1) \\ \implies \psi_{jk}(x) &= \varphi(2^{j+1}x - 2k) - \varphi(2^{j+1}x - 2k - 1) \end{aligned}$$

which is the desired result.

4.38

We have

$$f(x) = -2\phi(4x) + 4\phi(4x-1) + 2\phi(4x-2) - 3\phi(4x-3) \in V_2$$

which can be expressed as follows:

$$\begin{aligned} f(x) &= -(\phi(2x) + \psi(2x)) + 2(\phi(2x) - \psi(2x)) \\ &+ (\phi(2x-1) + \psi(2x-1)) - \frac{3}{2}(\phi(2x-1) - \psi(2x-1)) \\ &= \psi(2x) - 3\psi(2x) - \frac{1}{2}\phi(2x-1) + \frac{5}{2}\psi(2x-1) \\ &= \frac{1}{2}(\phi(x) + \psi) - 3\psi(2x) - \frac{1}{4}(\phi(x) - \psi(x)) + \frac{5}{2} - \psi(2x-1) \\ &= \frac{1}{4}\phi(x) + \frac{3}{4}\psi(x) + (-3\psi(2x) + \frac{5}{2}\psi(2x-1)) \end{aligned}$$

Now, we have that

$$\begin{aligned} &\frac{1}{4}\phi(x) \in V_0 \\ &\frac{3}{4}\psi(x) \in W_0 \\ &-3\psi(2x) + \frac{5}{2}\psi(2x-1)) \in W_1 \end{aligned}$$

4.39

We have that

$$f(x) = 2\varphi(4x) + 3\varphi(4x-1) + \varphi(4x-2) - 3\varphi(4x-3) \in V_2$$

which can be expressed as follows:

$$\begin{aligned}
f(x) &= (\varphi(2x) + \psi(2x)) + \frac{3}{2}(\varphi(2x) - \psi(2x)) + \\
&\quad \frac{1}{2}(\varphi(2x-1) + \psi(2x-1)) - \frac{3}{2}(\varphi(2x-1) - \psi(2x-1)) \\
&= \frac{5}{2}\varphi(2x) - \frac{1}{2}\psi(2x) - \varphi(2x-1) + 2\psi(2x-1) \\
&= \frac{1}{2}(\varphi(x) + \psi(x)) - \frac{1}{2}\psi(2x) - (\varphi(x) - \psi(x)) + 2\psi(2x-1) \\
&= \frac{1}{4}\varphi(x) + \frac{9}{4}\psi(x) + (-\frac{1}{2}\psi(2x) + 2\psi(2x-1))
\end{aligned}$$

Now, we have that

$$\left(\frac{1}{4}\varphi(x)\right) \in V_0 \quad \left(\frac{9}{4}\psi(x)\right) \in W_0 \quad \left(-\frac{1}{2}\psi(2x) + 2\psi(2x-1)\right) \in W_1$$

4.40

```
###########################
import numpy as np
from matplotlib import pyplot as plt

def sample(f, n):
    sampling = []
    for k in xrange(2**n+1):
        sampling.append(f(k/2.**n))
    return sampling

def mother_approximation(f, n):
    def psi(x):
        if x < .5 and x >= 0:
            val = 1.
            return val
        elif x < 1 and x >= .5:
            val = -1.
            return val
        else:
            val = 0.
            return val
```

```
    def daughter(x):
        value = 0.
        for k, sam in enumerate(sample(f, n)):
            value += sam * psi(2**n * x - k)
        return value

    return wn

def father_approximation(f, n):
    def phi(x):
        if x < 1 and x >= 0:
            val = 1.
            return val
        else:
            val = 0.
            return val

    def son(x):
        value = 0.
        for k, sam in enumerate(sample(f, n)):
            value += sam * phi(2**n * x - k)
        return value

    return fn


def test():
    f = lambda x: (np.sin(2.*np.pi*x - 5.))/(np.sqrt(np.abs(x - (np.pi/20.

    x = np.linspace(0,1,50)
    for l in xrange(1,11):
        print l
        fn = fn_approx(f, l)
        fn_x = []
        for val in x:
            fn_x.append(fn(val))

        plt.subplot(121)
        plt.plot(x, f(x))

        plt.subplot(122)
        plt.plot(x, fn_x)
        plt.show()
```

```
#Part 2

def coeff(f, n, sampling):
    c_k = []
    b_k = []
    for k in xrange(len(sampling)):
        c_k.append((f((2.*k)/(2.**n+1)) + f((2.*k + 1.)/(2.**n+1)))/2.)
        b_k.append((f((2.*k)/(2.**n+1)) - f((2.*k + 1.)/(2.**n+1)))/2.)

def wavelet_decomp(f, n, l):
    l_list = np.linspace(l, (n-1), (n-l)+1)
    wn_list = []
    for ll in l_list:
        wn_list.append(wn_approx(f, ll)
    fl = fn_approx(f, l)

    def fn(x):
        value = fl(x)
        for wn in wn_list:
            value += wn(x)
        return value

    plt.plot(fl(x))
    plt.plot(wn_list[0](x))
    plt.show()

    return fn

def test2():
    f = lambda x: (np.sin(2.*np.pi*x - 5.))/(np.sqrt(np.abs(x - (np.pi/20.
    n = 10
    ls = [0,1,2,3,4,5,6,7,8,9]
    x = np.linspace(0,1,50)
    for l in ls:
        wavelet_decomp(f, n, l)
```

This is different because it has incorporated the mother function as well.