CSE 344 : Computer Vision Homework 18

Name : Arka sarkar
Roll Number : 2018222

In [1]:

```
import numpy as np
from scipy import signal
x=np.array([[12 ,14 ,15 ,17],[34 ,56 ,73 ,32],[ 67 ,43 ,31 ,21],[32 ,31 ,43 ,56]])
w=np.array([[-1 , 0],[0,-1]])
b=0.3
dy= np.array([[-2, 3, 4],[-1, 5, 3],[2, 3, 4]])
```

Compute updated 'b' and 'w' in the current layer. [0.5 marks]

In [2]:

```
# Calculation db
db = np.sum(dy)
print("db : ", db)
```

```
db :  21
```

In [3]:

```
#Calculation dw
dw = signal.convolve2d(x,dy,mode = "valid")

print("dw : \n", dw)
```

```
dw :
 [[764 859]
 [970 911]]
```

In [4]:

```
#updating w and b
learning_rate = 0.1

w_up = w - learning_rate*dw
b_up = b - learning_rate*db

print("updated w :\n", w_up)
print("updated b :\n", b_up)
```

```
updated w :
 [[-77.4 -85.9]
 [-97.  -92.1]]
updated b :
 -1.8
```

Compute 'dy' for the next layer (in the backward direction) [0.5 marks]

In [5]:

```
#calculating dx
dy_0 = np.zeros((dy.shape[0]+2,dy.shape[1]+2 )) #padded dy
dy_0[1:-1,1:-1] = dy
w_prime = np.rot90(w,2)
dx = signal.convolve2d(dy_0,w_prime, mode = "valid")

print("dx : \n", dx)
```

```
dx :
 [[ 2. -3. -4.  0.]
 [ 1. -3. -6. -4.]
 [-2. -2. -9. -3.]
 [ 0. -2. -3. -4.]]
```

In [8]:

```
print("dy for next layer :\n", dx)
```

```
dy for next layer :
 [[ 2. -3. -4.  0.]
 [ 1. -3. -6. -4.]
 [-2. -2. -9. -3.]
 [ 0. -2. -3. -4.]]
```

In [ ]:

In [ ]: