

# Newton Raphson

April 9, 2015

```
In [3]: def newton_raphson(x_, f, df, epsilon=0.00001):
        n = 1
        while True:
            x = x_ - f(x_)/df(x_)
            if abs(x - x_) < epsilon:
                return x
            x_ = x
            print "Value of x after %s iterations is: %s" % (n, x_)
            n = n+1
            if n > 50:
                print "Divergent too many iterations"
```

```
In [4]: import numpy as np
        f = lambda x: np.cos(x)-3*x+1
        df = lambda x: -np.sin(x) - 3
        newton_raphson(1, f, df)
```

Value of x after 1 iterations is: 0.620015952247

Value of x after 2 iterations is: 0.607120658147

Value of x after 3 iterations is: 0.607101648145

Out[4]: 0.60710164810312273

In [ ]: