

Take-home Midterm

Massimo Di Pierro

February 9, 2013

The deadline for the project is Sun Feb 10, 2013. Each problem is worth 5 points for a total of 60 points out of 100 total points for the final grade. You must work independently. Plagiarism will be reported. Solutions that are very similar will be considered plagiarism. (It is ok to only prove convergence for $0 < x < 1$).

1 Part 1

1.1 Problem 1

Implement a Taylor expansion for $\log(1 + x)$ using Python or C++. Discuss convergence and remainder within the convergence radius. No need to prove it does not converge outside known convergence radius.

1.2 Problem 2

Explain why a divergent infinite series such as

$$\sum_{n=1}^{\infty} \frac{1}{n}$$

can have a finite sum in floating point arithmetic. At what point will the partial sum cease to change?

1.3 Problem 3

Consider the expression

$$\frac{1}{1-x} - \frac{1}{1+x}$$

For what range of values of x it is difficult to compute this expression accurately in floating point arithmetic? Provide an alternative form for the same expression which gives more accurate results.

1.4 Problem 4

Write a program in Python or C++ to solve the quadratic equation $ax^2 + bx + c = 0$ using the standard quadratic formula

$$x = (-b \pm \sqrt{b^2 - 4ac}) / (2a)$$

or the alternate formula

$$x = (2c) / (-b \pm \sqrt{b^2 - 4ac})$$

You should detect complex routes but no need to solve for complex values. Your program should be robust and, specifically, it should work for the following input values:

a	b	c
6	5	-4
6e154	5e154	-4e154
0	1	1
1	-1e-5	1
1	-4	4 - (1e-6)
1e-155	-1e155	1e155

1.5 Problem 5

Write a program in Python or C++ to compute the variance of a sequence $\{x_i\}$ using the two equivalent formulas:

$$\sigma^2 = \frac{1}{n} \sum_{i=0}^{i < n} (x_i - \bar{x})^2 \quad (1)$$

$$\sigma^2 = \frac{1}{n} \left(\sum_{i=0}^{i < n} x_i^2 - n\bar{x}^2 \right) \quad (2)$$

Prove that the two expressions are equivalent. Device an input series $\{x_i\}$ for which the two expressions produce different results numerically. Is one better than the other?

1.6 Problem 6

Here is a typical implementation a the Black-Scholes pricing formula:

```
def cdf(z):
    if z > 6.0:
        I = 1.0
    elif z < -6.0:
        I = 0.0
    else:
        a = fabs(z)
        b1 = 0.31938153
        b2 = -0.356563782
        b3 = 1.781477937
        b4 = -1.821255978
        b5 = 1.330274429
        p1 = 0.2316419
        c1 = 0.398923
        t = 1.0/(1.0+a*p1)
        I = 1.0 - c1*exp(-z*z/2)*((((b5*t+b4)*t+b3)*t+b2)*t+b1)*t
        if z<0:
            I = 1.0-I
    return I

def priceBS(S, X, r, sigma, t):
    sqrt_t = sqrt(t)
    d1 = (log(S/X)+r*t)/(sigma*sqrt_t)+0.5*sigma*sqrt_t
    d2 = d1 - sigma*sqrt_t
    c = S*cdf(d1)-X*exp(-r*t)*cdf(d2)
    return c
```

where `cdf` is an an auxiliary funciton (it approximate the cumulative distribution function for the gaussian distribution) and `priceBS` is the actual pricing algorithm. The inputs are:

- S : underlying stock price
- X : strike price
- r : daily expected rate of return for the stock
- σ : daily volatility for the stock
- t : time to expiration in days (can be smaller than 1).

The function returns the Black-Scholes price for a European call option.

S and X are dollar amounts. Will the result be more accurate or less accurate if S and X are expressed in cents? What if they are expressed in thousands of dollars?

Will the result be more accurate or less accurate if time is pressed in years instead of days (and accordingly r, σ are yearly return and volatility)?

Plot the output for $X = 10.0, r = 0.001, \sigma = 0.03, t = 90.0$ as function of S in range $[0, 20]$.

2 Part 2

2.1 Problem 7

Solve analytically the following system of linear equations in (x, y, z, t) as function of a :

$$x + y + z + t = 1 \quad (3)$$

$$2x + y + z - t = 2 \quad (4)$$

$$x + 3y + z - t = 3 \quad (5)$$

$$2x - y + 9z + 7t = a \quad (6)$$

$$(7)$$

For which values of a this system has one solution? For which values of a it has no solution? For which values of a it has infinite solutions? For which values of a the problem is ill conditioned? Explain your answers.

2.2 Problem 8

What is the inverse of the following matrix?

$$\begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & a & 1 & 0 \\ 0 & b & 0 & 1 \end{pmatrix}$$

How may this arise in computational practice?

2.3 Problem 9

Write an algorithm to efficiently invert any matrix A meeting the following conditions:

- The matrix is non-singular
- For each row, one and only one term is non-zero

You are not allowed to call the Gauss-Jordan algorithm. Explain why it works. Implement the algorithm in Python or C++. Discuss the speed of your algorithm compared with the Gauss-Jordan one.

2.4 Problem 10

If A is a $n \times n$ matrix and x is a n -vector, which of the following computations requires less work? Explain why.

$$y = (xx^T)A \tag{8}$$

$$y = x(x^T A) \tag{9}$$

$$\tag{10}$$

2.5 Problem 11

For a square matrix, what is it easier to compute the 1-norm or the 2-norm? Why?

2.6 Problem 12

Suppose A is a positive definite matrix. Show that A is non-singular. Show that the inverse of A is also positive definite.