TANA15 Numerical Linear Algebra

Fredrik Berntsson (fredrik.berntsson@liu.se)

Computer Exercise 3. Nonlinear Equations and Least Squares

1 General information

The programming assignments are intended to take more than two hours to complete. It is therefore important to be well prepared so that the computer sessions are used effectively. The assignment consists of a mixture of theoretical exercises and practical programming. A few of the exercises are based on custom Matlab programs or datasets that are provided in order to save time. These can be downloaded from the course website.

2 The Newton Method for Nonlinear Equations

Consider the nonlinear system of equations,

$$f(x) = 0, \qquad f : \mathbb{R}^n \mapsto \mathbb{R}]^n.$$

Given an approximation $x^{(k)} \approx x^*$ of the root we linearize the equation,

$$f(x) \approx f(x^{(k)}) + J_f(x^{(k)})(x - x^{(k)}) = 0,$$

The next iterate $x^{(k+1)}$ can thus be computed as,

$$J_f(x^{(k)})s^{(k)} = -f(x^{(k)}), \qquad x^{(k+1)} = x^{(k)}+)s^{(k)}.$$

This is the Newton method.

Exercise 2.1 Implement the Newton method using Matlab.

```
>> [ Xk , k ]=EqSolveNewton( f , Jf , X0 );
```

Use $\|s^{(k)}\| < 10^{-10}$, where $s^{(k)}$ is the Newton update, as a stopping criterion. $\square$.

Exercise 2.2 Solve the nonlinear system of equations

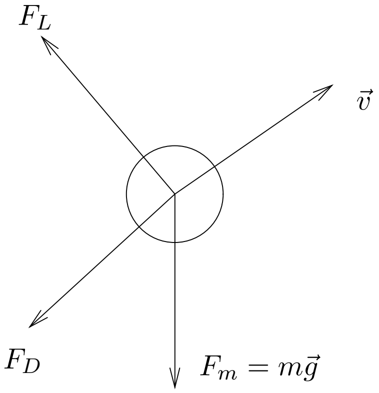
$$f(x) = \begin{pmatrix} (x_1+3)(x_2^3-7)+18 \\ \sin(x_2 \mathrm{e}^{x_1} - 1) \end{pmatrix} = 0,$$

using the starting point $x^{(0)} = (-0.5, 1.4)^T$. What is the expression for the Jacobian J_f?

The exact solution is $x^* = (01)^T$. Display the error at each iteration to verify the quadratic convergence. $\square$

Hint Use `inline` to define both the function $f(x)$ and the Jabobian $J_f(x)$.

3 Soccer Ball Trajectory



A Soccer ball that travels through air is subject to a *Drag force* F_D, *gravity pull* F_m, and a *Magnus force* F_L because of its spin. In this exercise we shall attempt to make the Soccer ball bounce at a certain spot by adjusting the initial velocity. The initial angular momentum ω that determines the Magnus force is assumed to be fixed.

Exercise 3.1 We want the ball to bounce at $P_e = (23, 6, 0)^T$; given the initial position $P_0 = (0, 0, 0)^T$. On the course Library there is a function `DisplayTrajectory` that can be used to visualize the trajectory given an initial velocity.

Find a starting guess for the initial velocity V_0 so that `EqSolveNewton` converges to the desired solution. The Jacobian has to be evaluated numerically using the function `BallJacobian`. The starting guess should be good enough so that the quadratic convergence starts immidiately.

How many function evaluations are needed to reach the stopping criterio n$\|S^{(k)}\| < 10^{-10}$? □

Evaluating the Jacobian is computationally expensive. An alternative is to use an approximate Jacobian $B_k \approx J_f(x^{(k)})$. Given two successive iterates $x^{(k+1)}$ and $x^{(k)}$ we update approximate Jacobian by a rank one matrix to satisfy the *secant equation*,

$$B_{k+1}(x^{(k+1)} - x^{(k)}) = f(x^{(k+1)}) - f(x^{(k)}).$$

The smallest such rank one update uv^T is given by the formula,

$$B_{k+1} = B_k + ((f(x^{(k+1)}) - f(x^{(k)}) - B_k s^{(k)})(s^{(k)})^T / \|s^{(k)}\|_2^2.$$

The resulting method is called *Broyden's Method.*

Exercise 3.2 Modify your Matlab function `EqSolveNewton` so that instead Broyden's Method, and starting Jacobian $B_0 = I$, is used. Use the same first guess for the initial velocity V_0 as previously. Compare the convergence rate with the previous results for the Newton method.

How many function evaluations $f(x)$ are needed now to reach the stopping rule $\|S^{(k)}\| < 10^{-10}$? □

4 Nonlinear Least Squares

We want to fit a circle to m given points $\{(x_i, y_i)\}$. The circle is parametrized by its center $z = (z_1, z_2)$ and radius r. A point is located on the circle if

$$r_i(u) = r_i(z_1, z_2, r) = \sqrt{(z_1 - x_i)^2 + (z_2 - y_i)^2} - r = 0.$$

We define a residual vector $r(u) = (r_1(u), \dots, r_m(u))^T$ and want to minimize,

$$f(u) = \|r(u)\|_2^2,$$

where u is the parameter vector $u = (z_1, z_2, r)$, and $r(x)$ maps $\mathbb{R}^3$ onto $\mathbb{R}^m$.

Exercise 4.1 Formulate the Gauss-Newton method for solving the nonlinear least squares problem above. Derive analytical expressions for the Jacobian matrix needed. □

On the course library there is a function `CircleData` that can be used to generate random points located close to a circle.

Exercise 4.2 Write a function,

```
>> u = GaussNewton( r , Jr , tol );
```

use $\|s^{(k)}\|_2 <$`tol`$= 10^{-10}$ as a stopping criterion. Experiment with different data sets, and starting approximations $u^{(0)}$. Is it difficult to get the iteration to converge?

Display computed circle and the data points for a couple of test cases. □