1 Newton corrector method

1.1 The formula

This method's concept is gotten from the concept of predictor corrector method. The process is

$$y \Rightarrow_{\mathbb{D}} y' \Rightarrow_{\mathbb{E}} y,$$

where $\mathbb{D}$ and $\mathbb{E}$ are operators to map y to y' and y' to y. Hence,

$$\mathbb{E} \circ \mathbb{D} \circ y = y \tag{1}$$

which is a recursion relation equation. In this process, $\mathbb{D}$ is the differential equation we want to solve, for example,

$$y' = 3y + 2x,$$

and $\mathbb{E}$ is the process of

$$y(l) = y(0) + \int_0^l \tilde{y}'(x)dx, \tag{2}$$

respectively. Where

$$\tilde{y}' \equiv \sum_{i=0}^{n} a_i x^i, \tag{3}$$

which is a nth order polynomial expansion for y' and if we substitute Eq. 3 into Eq. 2, we have

$$y(l) = y(0) + \sum_{i=0}^{n} \frac{a_i l^{i+1}}{i+1},$$

or in matrix form

$$y(l) = y(0) + \left(\frac{l}{1} \cdots \frac{l^{n+1}}{n+1} \right) \mathrm{M}^{-1}(s) \begin{pmatrix} y'(s) \\ \vdots \\ y'(n+s) \end{pmatrix} \equiv y(0) + E \begin{pmatrix} y'(s) \\ \vdots \\ y'(n+s) \end{pmatrix}, \tag{4}$$

where $M_{ij}(s) = (s+i-1)^{j-1}$ or in matrix form

$$\mathrm{M}(s) \begin{pmatrix} a_0 \\ \vdots \\ a_n \end{pmatrix} = \begin{pmatrix} s^0 & \cdots & s^n \\ \vdots & \ddots & \vdots \\ (n+s)^0 & \cdots & (n+s)^n \end{pmatrix} \begin{pmatrix} a_0 \\ \vdots \\ a_n \end{pmatrix} = \begin{pmatrix} y'(s) \\ \vdots \\ y'(n+s) \end{pmatrix},$$

which is a map from $\{a_i\}$ to $\{y'\}$.

Now, we have $\mathbb{D}$ and $\mathbb{E}$ operators, we still need y and the method of iteration. The initial y can be created by

$$y(l) = y(0) + l * \mathbb{D} \circ y(0),$$

and the method of iteration can be Newton's method which is

$$-\left(\mathbb{E}\circ\mathbb{D}\circ-1\right)y_n=\left[\frac{\delta\mathbb{E}}{\delta\mathbb{D}}\frac{\delta\mathbb{D}}{\delta y}-1\right]\left(y_{n+1}-y_n\right), \tag{5}$$

and its main purpose is to find $\left(\mathbb{E}\circ\mathbb{D}\circ-1\right)y=0$. How to find out $\frac{\delta\mathbb{E}}{\delta\mathbb{D}}$ and $\frac{\delta\mathbb{D}}{\delta y}$? They can be find out from the map of

$$\frac{\delta\mathbb{E}}{\delta\mathbb{D}}\delta y'=E\delta y'=\delta y,$$

which is derived from Eq. 4 and

$$\frac{\delta\mathbb{D}}{\delta y}\delta y=\delta y',$$

respectively.

In Eq. 5, we do not restrict y_n must be the value at one x point, we can set y_n to be of a set of $\{x\}$, so that we can use it to find out the y at beginning instead of R-K method.

1.2 $y(1\times1)$ example

For example, if we want to get the solution of $y'=0.1y$ and $y(0)=1$ case, and use $n=5$ polynomial to fit the solution, we need 6 x points for $\mathbb{E}$ operator. Here $\mathbb{D}\circ y=0.1*y=D*y$ and for first 6 points

$$\begin{aligned}\mathbb{E}:\begin{pmatrix}y(1)\\y(2)\\\vdots\\y(5)\end{pmatrix}&=\underbrace{\begin{pmatrix}1&\cdots&\frac{1}{6}\\\vdots&\ddots&\vdots\\5&\cdots&\frac{5^6}{6}\end{pmatrix}\begin{pmatrix}1&0&\cdots&0\\1&1&\cdots&1\\\vdots&&\ddots&\vdots\\1&5&\cdots&5^5\end{pmatrix}^{-1}}_{E}\begin{pmatrix}y'(0)\\y'(1)\\\vdots\\y'(5)\end{pmatrix}+y(0)\\&=E_1y'+E_0y'(0)+y(0).\end{aligned}\tag{6}$$

Hence, $\frac{\delta\mathbb{D}}{\delta y}=D=0.1$ and $\frac{\delta\mathbb{E}}{\delta\mathbb{D}}=E_1$, respectively.

Following are the details for each step.

- step 1: Provide the starting value $y(0)$, and $y'(0)$ can be derived from $y'(0)=\mathbb{D}\circ y(0)$. Here $y(0)=1$ and $y'(0)=0.1$.
- step 2: $y(l)=y(0)+l*y'(0)$.
- step 3: Check $|(1-\mathbb{E}\circ\mathbb{D})y|<\epsilon$, where $\epsilon\to0$, if it satisfies this relation, jump to step 5. Where

$$(1-\mathbb{E}\circ\mathbb{D}\circ)y=\begin{pmatrix}y(1)\\\vdots\\y(5)\end{pmatrix}-\left[E_1D\begin{pmatrix}y(1)\\\vdots\\y(5)\end{pmatrix}+E_0y'(0)+y(0)\right]. \tag{7}$$

```
E:\Job\my_study\predictor_corrector\test_1D.m                                    Page 1
2006年3月22日                                                              上午 11:32:13
%2006/03/20    test_1D y'=y/10

iterate=100; % # of iterate for Newton's method.
N=100; % # of points
n=10; %number of points for polynomial fitting
y_old=zeros(n-1,1);
y=zeros(N,1);
%% ///// Create the inital n points \\\\ %%
%1. provide initial value
y(1)=1;
%2. provide D
D=0.1;
D0=D;
%3. get E
[E,E0]=E_op_begin(n);
%4. initialize y
py=D*y(1);
%///// Predictor region \\\\\%
y_old=py*[1:(n-1)]'+y(1);
%\\\\\ Predictor region /////%
E0yp0y0=E0*py+y(1);
%///// Corrector region \\\\\%
for i0=1:iterate
  %5. -(ED-1)y_n \equiv dy
  dy=y_old-[E*D*y_old+E0yp0y0];
  if sqrt(dy'*dy) < 10*eps, break; end
  %6. DD = \delta E / \delta D * \delta D / \delta y -1
  DD=E*D-eye(n-1);
  y_old=inv(DD)*dy+y_old;
end
if i0==iterate, warning('too many times iterate for beginning of Newton-Corrector');end
%\\\\\ Corrector region /////%
y(2:n)=y_old;
%% \\\\\ Create the inital n points ///// %%

%% ///// Fit the higher points \\\\\ %%
py=D*y_old;
E=E_op(n);
E0=E(1:n-1);E1=E(n);
clear y_old D DD
for i0=(n+1):N
  %D0=
  E0yp0y0=E0*py+y(i0-1);
  y_old=(D0+1)*y(i0-1);  % initialize the data
  for i1=1:iterate
    dy=y_old-[E1*D0*y_old+E0yp0y0];
    if sqrt(dy'*dy) < 10*eps, break;end
    DD=E1*D0-1;
    y_old=dy/DD+y_old;
  end
  if i1==iterate, warning(['too many times iterate for ',num2str(i0),'th Newton-Corrector']);end
  y(i0)=y_old;
  %/// Renew py \\\%
  py_buf=py;
  py(1:n-2)=py_buf(2:n-1);
  py(n-1)=y(i0)*D0;
  %\\\ Renew py ///%
end
%% \\\\\ Fit the higher points ///// %%
```

Figure 1: The code for solving 1 dimensional y

- step 4: Use Newton's method as shown in Eq. 5 to get y_{n+1}, where $\frac{\delta\mathbb{E}}{\delta\mathbb{D}}\frac{\delta\mathbb{D}}{\delta y} - 1 = E_1 D - 1$. Because it is just a $b = Ax$ problem, we can get $y_{n+1} = y_n + A^{-1}b$ and input y_{n+1} for step 3.
- step 5: Output $y(1)$ to $y(5)$.

For $y(l)$, $l > 5$ case, we can treat them as conventional predictor-corrector method. In this method, $\mathbb{E}$ should be changed to be

$$\mathbb{E}: y(1) = y(0) + \underbrace{\left(1,\ \frac{1}{2}\cdots\frac{1}{6}\right)\begin{pmatrix} 1 & (-4) & \cdots & (-4)^5 \\ \vdots & & \ddots & \vdots \\ 1 & 0 & \cdots & 0 \\ 1 & 1 & \cdots & 1 \end{pmatrix}^{-1}}_{E}\begin{pmatrix} y'(-4) \\ y'(-3) \\ \vdots \\ y'(1) \end{pmatrix} \tag{8}$$

$$= E_0 y' + E_1 y'(1) + y(0),$$

and $\frac{\delta\mathbb{E}}{\delta\mathbb{D}} = E_0$. When you do the same steps as previous case, you will get $y(1)$ which in fact is $y(l)$ for the real solution.

Figure 1 is the matlab code for this case, in fact, if you change the $\mathbb{D}$ and $y(0)$, the code can be used to calculate other cases.

The subroutine related to this part are $newton\_corrector.m$, $M\_inv.m$, $E\_op.m$, $E\_op\_begin.m$, $Enk\_no\_int.m$. In fact, this method also can be used for m-dimension y and what you need to do is doing the following transfer:

$D \Rightarrow \bigoplus_{i=1}^{n} D_i$ for first n points fitting and $E \Rightarrow E \otimes I_m$ for all curve which are used in $newton\_corrector.m$.

1.3 Usage of $newton\_corrector.m$

Basically, its formula is

$$y = newton\_corrector(fun, y0, N, n),$$

where fun is the function name whose formula is $D = fun(x)$ and $y' = D(x)y$, $y0$ is the initial value of y, $x = 1 \rightarrow N$, $x_{m+1} - x_m = 1$ and n denotes how many points we used for the polynomial fitting, respectively.

You can test it by running $test.m$, it will show you two kinds of examples: the first one is for the solution of Dirac equation of hydrogen like atom without electron-electron interaction whose reference can be found in $http://www.physics.orst.edu/\sim allenlw/Ph65456/Media/PDFs/QM656.28.DiracHatom.pdf$, they have good explanation for this part, the second one is $y'' = -0.01y$ whose solution is $y = cos(0.1x)$, respectively.