Chapter Eight
Power Series

8.1 Types of functions

Mathematical functions can be divided into two categories:

a) Ordinary functions with finite number of terms: Which comprise all functions of one or more terms. The analytical evaluation of these functions gives exact results. The following are example of ordinary finite−terms functions:

$$y = x^2 + 3x - 1 \qquad y = \frac{x-1}{x+1} \qquad A = \sin(B) \qquad R = \ln(1+x)$$

b) Power series (Maclaurin series): In this category the function is composed of theoretically infinite number of terms that are added together. The added terms contain powers that are changed from term to term at a unique manner for each function. Power series are usually a second representation of ordinary functions but they give approximate results. The error between the ordinary and series forms of a function can be minimized by increasing the series terms considered in the evaluation. Power series of use must converge to a finite value, in other words, the value of their successive terms must decrease from term to term. The divergent series result in an infinite value when evaluated. The following are examples of power series with their ordinary forms:

$$\left.\begin{aligned}
y &= \sum_{n=0}^{\infty} \frac{x^n}{n!} = 1 + x + \frac{x^2}{2!} + \frac{x^3}{3!} + \frac{x^4}{4!} + \ldots\ldots + \frac{x^\infty}{\infty!} \quad = e^x \\
\mathrm{y} &= \sum_{n=0}^{\infty} \frac{(-1)^n x^{2n+1}}{(2n+1)!} = x - \frac{x^3}{3!} + \frac{x^5}{5!} - \frac{x^7}{7!} + \ldots\ldots + \frac{x^\infty}{\infty!} = \sin(x) \\
\mathrm{y} &= \sum_{n=0}^{\infty} \frac{(-1)^n x^{2n}}{(2n)!} = 1 - \frac{x^2}{2!} + \frac{x^4}{4!} - \frac{x^6}{6!} + \ldots\ldots + \frac{x^\infty}{\infty!} = \cos(x)
\end{aligned}\right\}$$

Convergent Series:
at x = finite value
y = finite result

$$y = \sum_{n=0}^{\infty} x^n = 1 + x + x^2 + x^3 + x^4 + \ldots\ldots + x^\infty$$

Divergent Series (for x>1):
at x = finite value
y = infinite result

8.2 Computation error:

It is defined as the absolute value of the difference between exact and approximate values divided by the exact value, as follows:

$$error = \left| \frac{exact - approximated}{exact} \right|$$

Ex. 8.1 Write MATLAB program to evaluate the following power series which is the other form of the function $y=e^x$. Take five terms of the series and evaluate the percentage error to the exact value. Consider any entry of x:

$$y = \sum_{n=0}^{\infty} \frac{x^n}{n!} = e^x$$

Sol.

```
clear,clc
x=input(' x = ');
ye=exp(x); y=0;
for n=0:4
    y=y+x^n/factorial(n);
end
err=abs((ye-y)/ye);
fprintf(' x            = %2.0f \n',x)
fprintf(' y (exact)    = %10.8f \n',ye)
fprintf(' y (approx.) = %10.8f \n',y)
fprintf(' error (%%)    = %4.2f%% \n',err*100)
```

Ex. 8.2 Write MATLAB program to evaluate the series of sine and cosine as given below for any value of x (in radians). Consider 8 term of each series and calculate the value of associated errors:

$$y = \sum_{n=0}^{\infty} \frac{(-1)^n x^{2n+1}}{(2n+1)!} = \sin(x)$$

$$y = \sum_{n=0}^{\infty} \frac{(-1)^n x^{2n}}{(2n)!} = \cos(x)$$

Sol.

```
clear,clc
xd=input('Enter an Angle in Degrees = ');
x=xd*pi/180; y=0; z=0;
ye=sin(x);ze=cos(x);
for n=0:7
    y=y+(-1)^n*x^(2*n+1)/factorial(2*n+1);
    z=z+(-1)^n*x^(2*n)/factorial(2*n);
end
err1=abs((ye-y)/ye); err2=abs((ze-z)/ze);
fprintf('\n')
fprintf(' function x(Deg.) y(exact) y(approx.) error \n')
fprintf('--------------------------------------------\n')
fprintf('   sin(x)   %4.2f   %6.4f   %6.4f   %9.7f...
                                   \n',[xd,ye,y,err1])
fprintf('   cos(x)   %4.2f   %6.4f   %6.4f   %9.7f...
                                   \n',[xd,ze,z,err2])
```

Ex. 8.3 Write MATLAB program to print a table displaying the decrease of computational error with the increase of the number of terms considered in evaluating the following power series for any value of x:

$$y = \sum_{n=0}^{\infty} \frac{(1+x)^{1+n}}{n!}$$

Sol.

```
clear,clc
x=input(' x = ');
y=0;
for a=0:12
  y_old=y;
  y=0;
    for n=0:a
        y=y+(1+x)^(1+n)/factorial(n);
    end
  err=abs((y-y_old)/y);
  R=[a,y_old,y,err];
  fprintf(' %4.0f %8.3f %8.3f %8.3f  \n',R)
end
```

Run:

x = 2

0	0.000	3.000	1.000
1	3.000	12.000	0.750
2	12.000	25.500	0.529
3	25.500	39.000	0.346
4	39.000	49.125	0.206
5	49.125	55.200	0.110
6	55.200	58.238	0.052
7	58.238	59.539	0.022
8	59.539	60.027	0.008
9	60.027	60.190	0.003
10	60.190	60.239	0.001
11	60.239	60.252	0.000
12	60.252	60.256	0.000