HW-1

1. Taylor's polynomial approximating a function is defined as follows

$$P_n(x) = f(a) + \sum_{j=1}^{n} \frac{f^{(j)}(a)(x-a)^j}{j!}$$

$$R_n(x) = \frac{(x-a)^{n+1}}{(n+1)!} f^{(j)}(\mu), \qquad \alpha \le x \le \beta \text{ , and } a \le \mu \le x.$$

$$f(x) = P_n(x) + R_n(x).$$

Use the Matlab symbolic computation functions: `syms x; diff(f,x,j),` to derive the derivatives manually first and then the Taylor polynomials for the following functions:

i. $f(x) = e^x \sin(x)\, , a = 0, n = 5.$

ii. $f(x) = \log\left(\frac{1+x}{1-x}\right) = \log(1+x) - \log(1-x)[HINT], a = 0, n = 5.$

Can you see the patterns and show the general Taylor's polynomial for arbitrary n for both I and ii.?

Can you compute $\log(3)$ using the above general polynomial for (ii) within an accuracy of 10^{-5}? What is the n in this case?

2. The following Matlab script and function can implement the Taylor approximation using a combination of symbolic computation and classical programming and writes the results in a file:

```
z=input('z=');
a=input('a=');
for n=1:10
    [ result ,error] = taylor( z,a,n );
    v(n)=result;e(n)=error;
end
x=1:n;
disp( '          n        n(1)       error')
disp([x' v' e'])
fid=fopen('expsin.txt','w');
fprintf(fid,'%s\n',' n     Pn(1)          error');
fprintf(fid,'%2u %14.10f %14.10f\n',[x;v;e]);
fclose(fid);

function [ result ,error] = taylor( z,a,n )
syms x real;
f=exp(x)*sin(x);
sum=subs(f,'x',a);
prod=1;
for j=1:n
    prod=prod*(z-a)/j;
    sum=sum+prod*subs(diff(f,x,j),'x',a);
end
format long
result=double(sum);
error=double(abs(result-subs(f,'x',z)));
end
```

HW-1

i. `Run the programs for both functions given  in 1 to verify your results derived manually.`

ii. One issue of the program above is that it uses `subs(f,'x',z)` to determine the $error$. We need to replace this process with something computable since in general you will not know your function but rather its polynomial. We could use $error(n,m) = |P_n(z) - P_m(z)|, m > n$. Is this a good approximation for the error? Evaluate for different $m > n$ by comparing with the exact error and write down your conclusions.