MATH661 HW08 - Linear operator approximation

Posted: 10/25/23
Due: 11/06/23, 11:59PM

The basic idea in linear operator approximation is to apply the exact operator to an approximation of the input. These exercises explore and reinforce this concept.

1 Track 1

1. Use Taylor series expansions to verify the approximations

$$f'(t) \cong \frac{1}{12h}[-f(t+2h)+8f(t+h)-8f(t-h)+f(t-2h)], \tag{1}$$

$$f'(t) \cong \frac{1}{2h}[-3f(t)+4f(t+h)-f(t+2h)]. \tag{2}$$

Determine the error term. Construct the polynomial approximant $p_n(t) \cong f(t)$ whose derivative leads to the above formula. Conduct a convergence study as $h \to 0$ for $f(t) \in \{\sin(\pi t/4), e^{10t}, e^{-10t}\}$ at $t_0 = 1$, and compare the observed order of convergence with the theoretical estimate.

Solution. For hand computation, organize Taylor series expansions into a table with c the coefficients from the finite difference formula. Observe that Taylor series terms of even powers in h cancel out, and those of odd powers from $-kh$ simply double those from kh parts of the expansions. The power of h multiplying $f^{(k)}$ is h^{k-1}. Observing and exploiting such symmetries reduces the number of calculations considerably.

k				1		3		5
term	c	$2c$	$f'(t)$	$2cf'(t)$	$f'''(t)$	$2cf'''(t)$	$f^{(5)}(t)$	$2cf^{(5)}(t)$
$f(t+2h)$	$-\frac{1}{12}$	$-\frac{1}{6}$	2	$-\frac{1}{3}$	8	$-\frac{4}{3}$	32	$-\frac{16}{3}$
$f(t+h)$	$\frac{8}{12}$	$\frac{4}{3}$		$\frac{4}{3}$	1	$\frac{4}{3}$		$\frac{4}{3}$
Sum				1		0		-4
Sum$/k!$				1		0		$-\frac{1}{30}$

Deduce leading-order error

$$\varepsilon_1 = -\frac{h^4 f^{(5)}(\xi)}{30}.$$

This can be verified in Mathematica.

```
Series[(-f[t+2h] + 8 f[t+h] - 8 f[t-h] + f[t-2h]) / (12 h), {h, 0, 5}]

f′[t] - 1/30 f^(5)[t] h^4 + O[h]^6
```

Set origin at $t=0$. Formula (1) contains data $\mathcal{D} = \{(kh, f_k), k=-2,-1,1,2\} \bigcup \{0, f_0'\}$ with $f_k = f(kh)$.

for a total of 5 conditions that can be satisfied by polynomial $p(t)$ of degree $n \geqslant 4$.

$$p(-2h) = f_{-2},\, p(-h) = f_{-1},\, p(h) = f_1,\, p(2h) = f_2,$$

$$p'(0) = f_0' = \frac{1}{12h}[-f_2 + 8f_1 - 8f_{-1} + f_{-2}].$$

This exercise highlights use of the Lagrange form of a Hermite interpolating polynomial with differing types of information at each node. The Lagrange basis functions

$$a_i(t) = a_i(\alpha h) = [1 - 2(\alpha - i)h\ell_i'(ih)]\ell_i^2(\alpha h),\ b_0(t) = b_0(\alpha h) = \alpha h\, \ell_0^2(\alpha h),$$

$$\ell_i(t) = \ell_i(\alpha h) = \prod_{k=-2,k\neq 0,i}^{2} \frac{\alpha h - kh}{ih - kh} = \prod_{k=-2,k\neq 0,i}^{2} \frac{\alpha - k}{i - k}, i = -2, -1, 1, 2,$$

$$\ell_0(t) = \ell_0(\alpha h) = \prod_{k=-2,k\neq 0}^{2} \frac{\alpha h - kh}{(-kh)} = \frac{1}{4}(\alpha^2 - 1)(\alpha^2 - 2)$$

satisfy the desired conditions leading to the polynomial

$$p(t) = p(\alpha h) = f_0'\, b_0(\alpha h) + \sum_{i=-2,i\neq 0}^{2} f_i\, a_i(\alpha h).$$

A similar analysis can be applied to the second formula giving the error term

$$\varepsilon_2 = -\frac{h^2 f^{(3)}(\xi)}{3}.$$

Steps for the convergence study:

- Define the finite difference derivative approximations
- Define the specified study functions
- Define a function to construct the convergence plot by: evaluating the error

  $$u_k = \lg(e_k) = \lg|D_{h/2} f - D_h f| \cong p - q \lg h_k$$

  for $h_k = h_0 2^{-k}$, $k = 1, \ldots, N$, extracting the order of convergence p by least squares (linear regression)

  $$\boldsymbol{A} = [\ 1 \quad \lg h\ ] = \begin{bmatrix} 1 & 1 \\ 1 & 2 \\ \vdots & \vdots \\ 1 & N \end{bmatrix};\ \min_{(p,q)} \left\| \boldsymbol{A} \begin{bmatrix} p \\ q \end{bmatrix} - \boldsymbol{u} \right\| \Rightarrow \begin{bmatrix} p \\ q \end{bmatrix} = \boldsymbol{A} \backslash \boldsymbol{u},$$

  and plotting the observed behavior.

```
∴ function conv(h0,t,f,D,N,fname,figname)
    df=zeros(N+1,1); u=zeros(N,1); h=h0; lgh=zeros(N+1,1)
    for k=1:N+1
      h=h/2; lgh[k]=log10(h); df[k]=D(t,f,h);
    end
    A=ones(N,2); x=-lgh[1:N]; A[:,2]=x;
    u[1:N]=log10.(abs.(df[2:N+1]-df[1:N]));
    q = floor((A\u)[2]*1000)/1000;
    plot(x,u,"o"); grid("on"); title(figname*"␣q="*string(q))
    xlabel(L"$-\lg(h)$"); ylabel(L"\lg(e_k)")
    savefig(fname)
    return q
  end;
```

```
∴
```

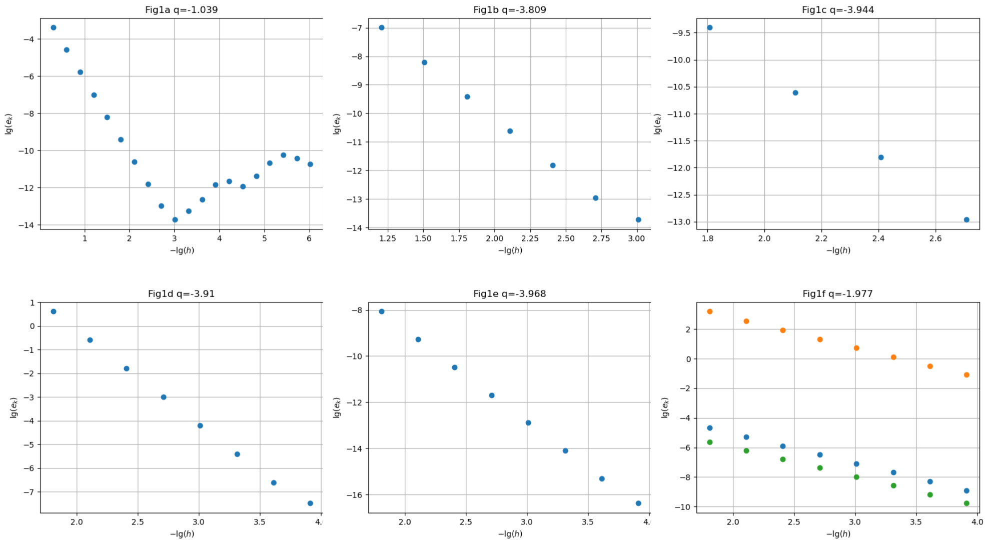


Figure 1. Convergence study results. Top row for $f_1(t) = \sin(\pi t/4)$: (1a) Results exhibit loss of precision as h decreases, and $q \cong -1$ is affected by error increase for small h; (1b) The first few step sizes after $h_0 = 1/8$ are not yet in the asymptotic regime, hence the observed order of convergence is $q = 3.81$, less than the theoretical $\mathcal{O}(h^4)$ order. (1c) The asymptotic regime is reached at $h_0 = 1/32$, and the floating point precision is sufficient up to $h = h_0/2^4$ to obtain $q = 3.94$, close to the theoretical order. Second row: (2a) Again for $h_0 = 1/32$ to $h = h_0/2^8$ the observed order for $f_2(t) = \exp(10t)$ is $q = 3.9$, close to the theoretical prediction. The amplification of $\|f_2\|_\infty \approx e^{10} \approx 2.2 \times 10^4$ allows more accurate significant digits in floating point, but the overall precision is small (2b) The same h range for f_3 with $\|f_3\| \approx 4 \times 10^{-5}$ not only achieves predicted order of accuracy, $q = 3.97$, but also much better accuracy with machine precision $\epsilon \approx 10^{-16}$ achieved when $h = 1/32/2^8 \cong 10^{-4}$. (2c) Second finite difference formula results for f_1 (●), f_2 (●), f_3 (●). The observed order of conergence is close to the theoretical $\mathcal{O}(h^2)$ in all cases, albeit with different overall errors.

```
∴ figdir=homedir()*"/courses/MATH661/homework/H08/";
∴ clf(); h0=1; t0=1; conv(h0,t0,f1,D1,20,figdir*"H08Fig1a.png","Fig1a");
∴ clf(); h0=1/8; conv(h0,t0,f1,D1,7,figdir*"H08Fig1b.png","Fig1b");
∴ clf(); h0=1/32; conv(h0,t0,f1,D1,4,figdir*"H08Fig1c.png","Fig1c");
∴ clf(); h0=1/32; conv(h0,t0,f2,D1,8,figdir*"H08Fig1d.png","Fig1d");
∴ clf(); h0=1/32; conv(h0,t0,f3,D1,8,figdir*"H08Fig1e.png","Fig1e");
∴ clf(); h0=1/32; conv(h0,t0,f1,D2,8,figdir*"H08Fig1f.png","Fig1f");
∴ h0=1/32; conv(h0,t0,f2,D2,8,figdir*"H08Fig1f.png","Fig1f");
∴ h0=1/32; conv(h0,t0,f3,D2,8,figdir*"H08Fig1f.png","Fig1f");
∴
```

2. As above for

$$f''(t) \cong \frac{1}{12h^2}[-f(t+2h) + 16f(t+h) - 30f(t) + 16f(t-h) - f(t-2h)],$$

$$f'''(t) \cong \frac{1}{h^3}[f(t+3h) - 3f(t+2h) + 3f(t+h) - f(t)].$$

Solution. Symbolic computation in Mathematica readily gives $\mathcal{O}(h^4)$, $\mathcal{O}(h)$ behavior respectively.

```
Series[(-f[t + 2 h] + 16 f[t + h] - 30 f[t] + 16 f[t - h] - f[t - 2 h]) / (12 h^2), {h, 0, 5}]
```

$$f''[t] - \frac{1}{90} f^{(6)}[t]\, h^4 + O[h]^6$$

```
Series[(f[t + 3 h] - 3 f[t + 2 h] + 3 f[t + h] - f[t]) / (h^3), {h, 0, 3}]
```

$$f^{(3)}[t] + \frac{3}{2} f^{(4)}[t]\, h + \frac{5}{4} f^{(5)}[t]\, h^2 + \frac{3}{4} f^{(6)}[t]\, h^3 + O[h]^4$$

- Define the finite difference derivative approximations

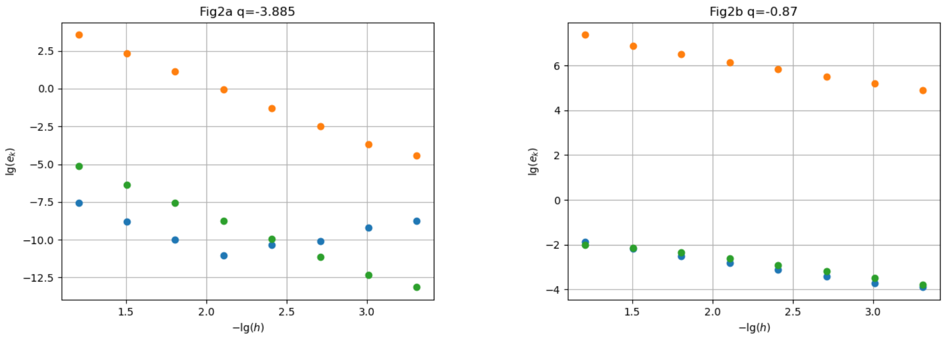
- 

Figure 2. Convergence study results. Left: f'' formula results for f_1 (●), f_2 (●), f_3 (●), fourth order achieved for certain h ranges. Right: f''' formula results for f_1 (●), f_2 (●), f_3 (●), with $q \lesssim 1$

3. Construct a recursive function `RecInt(a,b,err,f,Q)` that has arguments scalars a, b, err and functions f, Q and approximates

$$I(f) = \int_a^b f(t)\,\mathrm{d}t$$

through repeated application of quadrature rule $Q(f,a,b)$ according to the algorithm

Algorithm Recursive quadrature

RecInt(a,b,err,f,Q)
$c = a + (b - a)/2$
$Q_{\mathrm{ab}} = Q(f,a,b)$; $Q_{\mathrm{ac}} = Q(f,a,c)$; $Q_{\mathrm{cb}} = Q(f,c,b)$
$e = |Q_{\mathrm{ac}} + Q_{\mathrm{cb}} - Q_{\mathrm{ab}}| / |Q_{\mathrm{ac}} + Q_{\mathrm{cb}}|$
if $e < \mathrm{err}$
    return $Q_{\mathrm{ac}} + Q_{\mathrm{cb}}$
else
    return RecInt(a,c,err,f,Q)+RecInt(c,b,err,f,Q)

Test the recursive integration procedure with trapezoid, Simpson, and Gauss-Legendre rules of orders 2,3 on the integral

$$\int_{-1}^{1} \cos\left(\frac{1}{t}\right).$$

For each case, present plots of the integrand and the evaluation points used in the recursive quadrature algorithm. Construct convergence plots by executing the algorithm for various error thresholds ε_k and recording the number of evaluation points n_k. Plot $(\log n_k, \log \varepsilon_k)$ and comment on whether the observed order of convergence is that predicted by theoretical quadrature error estimates.

Solution. Define:

- Recursive quadrature procedure

```
∴ function RecInt(a,b,eps,f,Q,level)
    global nf,nmax,lmax
    c=a+(b-a)/2; Qab=Q(a,b,f); Qac=Q(a,c,f); Qcb=Q(c,b,f)
    new=Qac+Qcb; old=Qab;
    if nf > nmax/2 return new end
    if level > lmax return new end
    err=abs((new-old)/new)
    if err<eps
      return new
    else
      return RecInt(a,c,eps,f,Q,level+1)+RecInt(c,b,eps,f,Q,level+1)
    end
  end;
∴
```

Quadrature rules:

- Trapezoid

$$\int_a^b f(t)\,\mathrm{d}t = \frac{b-a}{2}[f(a)+f(b)].$$

```
∴ function trapezoid(a,b,f)
    return 0.5*(b-a)*(f(a)+f(b))
  end;
∴
```

- Simpson

$$\int_a^b f(t)\,\mathrm{d}t = \frac{b-a}{6}\left[f(a)+4f\left(\frac{a+b}{2}\right)+f(b)\right].$$

```
∴ function simpson(a,b,f)
    return (b-a)/6 * (f(a)+4*f(0.5*(a+b))+f(b))
  end;
∴
```

- Gauss-Legendre 2

$$\int_{-1}^1 f(t)\,\mathrm{d}t = f(-\sqrt{3})+f(\sqrt{3}).$$

```
∴ function gl2(a,b,f)
    s=(b-a)/2; z(t)=s*(t+1)+a; s3=sqrt(1/3);
    return s*(f(z(-s3))+f(z(s3)))
  end;
∴
```

- Gauss-Legendre 3

$$\int_{-1}^1 f(t)\,\mathrm{d}t = \frac{5}{9}f\left(-\sqrt{\frac{3}{5}}\right)+\frac{8}{9}f(0)+\frac{5}{9}f\left(\sqrt{\frac{3}{5}}\right)$$

```
∴ function gl3(a,b,f)
    s=(b-a)/2; z(t)=s*(t+1)+a; s35=sqrt(3/5); w1=5/9; w0=8/9
    return s*(w1*f(z(-s35)) + w0*f(z(0)) + w1*f(z(s35)))
  end;
```

```
∴
```

- Define the integrand. Each time the integrand is called record the evaluation point and increment a counter of function evaluations. The integrand is singular at $t=0$, but over an interval of measure zero in the limit. Hence setting the integrand value to zero at $t=0$ does not affect the integral value. Also define a simpler integrand g for overall testing.

```
∴ function f(t)
    global nf,tvals,fvals
    if abs(t)<1.0e-6
      fval=0.
    else
      fval=cos(1/t)
    end
    nf = nf+1; tvals[nf]=t; fvals[nf]=fval
    return fval
  end;
```

```
∴ function g(t)
    global nf,tvals,fvals
    fval = cos(t)
    nf = nf+1; tvals[nf]=t; fvals[nf]=fval
    return fval
  end;
```

```
∴
```

  - Initialize function evaluation counter, values, call the routine for simple integrand

$$\int_0^{\pi/2} \cos(t)\,\mathrm{d}t = 1$$

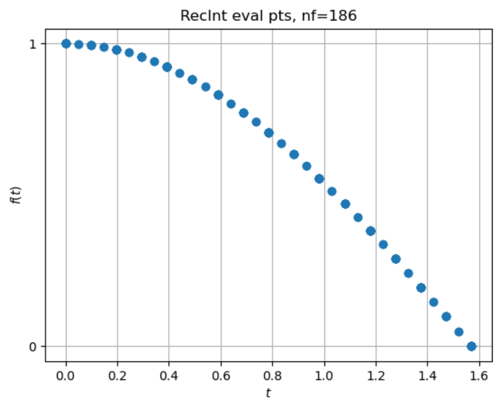


Figure 3. For integrand $g(t)=\cos(t)$ function evaluation points are close to an equidistant interval partition. Three accurate digits ($\varepsilon < 10^{-3}$) are attained with n_f=186,27,18,9 function evaluations, highlighting the effectiveness of high-order quadrature.

- Initialize function evaluation counter, values, call the routine for singular integrand

$$\int_1^{-1} \cos(1/t)\,\mathrm{d}t = 2-\pi+\int_0^1 \frac{\sin t}{t}\mathrm{d}t \approx -0.1688$$

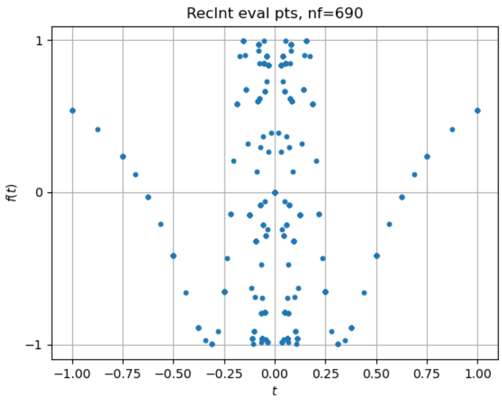


Figure 4. For integrand $f(t) = \cos(1/t)$ function evaluation points are clustered in regions of rapid variation of the function. Within a limit on the number of function evaluation $n_f \cong 5 \times 10^5$, only the GL3 quadrature gives a value of -0.16 with two accurate digits. This example highlights both recursive quadrature and the need for specialized numerical quadrature procedures for highly oscillatory or singular integrands.

```
∴ global nf,nmax,lmax
∴ nf=0; nmax=100000; lmax=8; tvals=zeros(nmax,1); fvals=zeros(nmax,1);
∴ Qg=RecInt(-1,1,10.0^(-1),f,trapezoid,0); [nf Qg]
```

$$[\ 690.0 \quad -0.144658181649111\ 76\ ] \tag{7}$$

```
∴ clf(); plot(tvals[1:nf],fvals[1:nf],".");
∴ xlabel(L"$t$"); ylabel(L"$f(t)$"); grid("on");
∴ title("RecInt␣eval␣pts,␣nf="*string(nf));
∴ figdir=homedir()*"/courses/MATH661/homework/H08/";
∴ savefig(figdir*"Fig3fT.png");
∴ nf=0; nmax=1000000; lmax=32; tvals=zeros(nmax,1); fvals=zeros(nmax,1);
∴ Qg=RecInt(-1,1,10.0^(-2),f,trapezoid,0); [nf Qg]
```

$$[\ 500058.0 \quad -0.1574321351531846\ ] \tag{8}$$

```
∴ nf=0; nmax=1000000; lmax=32; tvals=zeros(nmax,1); fvals=zeros(nmax,1);
∴ Qg=RecInt(-1,1,10.0^(-2),f,simpson,0); [nf Qg]
```

$$[\ 500067.0 \quad -0.37476168010680994\ ] \tag{9}$$

```
∴ nf=0; nmax=1000000; lmax=32; tvals=zeros(nmax,1); fvals=zeros(nmax,1);
∴ Qg=RecInt(-1,1,10.0^(-2),f,gl2,0); [nf Qg]
```

$$[\ 500046.0 \quad -0.4505206896512667\ ] \tag{10}$$

```
∴ nf=0; nmax=1000000; lmax=32; tvals=zeros(nmax,1); fvals=zeros(nmax,1);
∴ Qg=RecInt(-1,1,10.0^(-2),f,gl3,0); [nf Qg]
```

$$[\ 500103.0 \quad -0.16578576294469366\ ] \tag{11}$$

```
∴
```

2 Track 2

1. Use the finite difference expressions of the derivative

$$\frac{\mathrm{d}}{\mathrm{d}t} = \frac{1}{h}\left(\Delta - \frac{1}{2}\Delta^2 + \ldots\right) = \frac{1}{h}\left(\nabla + \frac{1}{2}\nabla^2 + \ldots\right) = \frac{1}{h}\left(\delta - \frac{\delta^3}{24} + \cdots\right)$$

to obtain the approximations.

$$f'(t) \cong \frac{1}{12h}[-f(t+2h)+8f(t+h)-8f(t-h)+f(t-2h)],$$

$$f'(t) \cong \frac{1}{2h}[-3f(t)+4f(t+h)-f(t+2h)].$$

Conduct a convergence study as $h \to 0$ for $f(t) \in \{\sin(\pi t/4), e^{10t}, e^{-10t}\}$ at $t_0 = 1$, and compare the observed order of convergence with theoretical estimates. How do the three functions differ, and what effect does this have on derivative approximation?

Solution. See Track 1 above for the standard convergence study. The Track 2 aspect of this exercise is to highlight how research problems arise.

The results

$$\frac{\mathrm{d}}{\mathrm{d}t} = \frac{1}{h}\ln(I+\Delta) = -\frac{1}{h}\ln(I-\nabla) = \frac{2}{h}\operatorname{arcsinh}\left(\frac{\delta}{2}\right),$$

$$\Delta = E - I, \nabla = I - E^{-1}, \delta = E^{1/2} - E^{-1/2}$$

indicate multiple ways to express the differentiation operator $D = \mathrm{d}/\mathrm{d}t$ in terms of the translation operator E. One consideration in constructing finite difference approximations of a derivative is the number of function sample points, known as the stencil size. For example, a fourth order accurate formula is obtained with a stencil size of 5 from $hD = \ln(I+\Delta)$, using sample points $t + kh$, $k = 0, 1, \ldots, 4$. The formula

$$\frac{\mathrm{d}}{\mathrm{d}t}f(t) \cong \frac{1}{12h}(-E^2 + 8E - 8E^{-1} + E^2)f(t),$$

is of fourth order accuracy, but has a stencil size of 4. This suggests exploring what additional expressions of D in terms of functions of $E, \Delta, \nabla, \delta$ might be found. This is also the typical research scenario: when a result does not conform to known theory, seek extensions of the theory.

Carry out calculations to obtain the requested approximation of the derivative in terms of finite differences

$$D = \frac{\mathrm{d}}{\mathrm{d}t} = \frac{(E - E^{-1})}{12h}[8I - (E + E^{-1})] = \frac{1}{2}\left[\frac{1}{h}\left(\Delta - \frac{1}{6}\Delta^2\right) + \frac{1}{h}\left(\nabla + \frac{1}{6}\nabla^2\right)\right].$$

This suggests an average of one-sided finite differences, truncated to first two terms,

$$D = \frac{1}{2}(F + B), F = \frac{1}{h}\left(\Delta - \frac{1}{6}\Delta^2 + \cdots\right), B = \frac{1}{h}\left(\nabla + \frac{1}{6}\nabla^2 + \cdots\right).$$

Take an additional research step and ask how the above may be extended. Can a function be found from the first few terms of its power series? Consider the following Mathematica result relevant for $hD = \ln(I - \nabla)$.

```
FindGeneratingFunction[{1, 1/2, 1/3, 1/4}, t]
```

$$-\frac{\text{Log}[1-t]}{t}$$

The generating function has been found using only 4 terms. Extra credit (4 pts). Can you find generating functions for F, B?

2. As above for

$$f''(t) \cong \frac{1}{12h^2}[-f(t+2h)+16f(t+h)-30f(t)+16f(t-h)-f(t-2h)],$$

$$f'''(t) \cong \frac{1}{h^3}[f(t+3h)-3f(t+2h)+3f(t+h)-f(t)].$$

Use the series products

$$\frac{\mathrm{d}^2}{\mathrm{d}t^2} = \frac{\mathrm{d}}{\mathrm{d}t}\frac{\mathrm{d}}{\mathrm{d}t} = \frac{1}{h^2}\left(\Delta - \frac{1}{2}\Delta^2 + \ldots\right)\left(\Delta - \frac{1}{2}\Delta^2 + \ldots\right).$$

Solution. As above.

3. Romberg integration is a combination of trapezoid quadrature over decreasing subintervals and Aitken extrapolation. Implement Romberg integration and test on

$$\int_0^1 e^t \cos(\pi t)\,\mathrm{d}t.$$

Present a convergence sutdy. What is the observed order of convergence?

Solution.