Catam Project Report
PartII Additional Projects
(July 2020 Edition)

Freud Fan

19/03/2021

1 Padé Approximants

1.1 Question 1

$f_1(x) = (1+x)^{1/2}$ is analytic at x=0, hence has taylor expansion with

$$\begin{aligned} f_1(0) &= 1,\ f_1'(0) = \frac{1}{2}(1+x)^{-1/2}\big|_0 = \frac{1}{2},\ f_1''(0) = -\frac{1}{4}(1+x)^{-3/2}\big|_0 = -\frac{1}{4} \\ f_1^{(k)}(0) &= \frac{1}{2}(-\frac{1}{2})(-\frac{3}{2})...(-\frac{2k-3}{2}) = \frac{(2k-3)!(-1)^{k-1}}{(k-2)!2^{2k-2}} \quad for \quad k \geq 2 \\ &= \frac{(2k-1)!(-1)^{k-1}}{(k-1)!(2k-1)2^{2k-1}} \quad for \quad k \geq 1, \end{aligned}$$

i.e.,

$$f_1(x) = 1 + \sum_{k=1}^{\infty} \binom{2k-1}{k} \frac{(-1)^{k-1}}{(2k-1)2^{2k-1}}\, x^k \tag{1}$$

around x=0. Coefficients

$$c_k = \begin{cases} 1 & k = 0 \\ \binom{2k-1}{k} \frac{(-1)^{k-1}}{(2k-1)2^{2k-1}} & k \geq 1 \end{cases}$$

To find the radius of convergence (r.o.c), use ratio test, the power series converge if

$$\lim_{k\to\infty} \left|\frac{c_{k+1}x^{k+1}}{c_k x^k}\right| = \lim_{k\to\infty} \left|\frac{(2k+1)!k!(k-1)!(-1)^k(2k-1)2^{2k-1}}{(2k-1)!(k+1)!k!(-1)^{k-1}(2k+1)2^{2k+1}}x\right| = \lim_{k\to\infty}\left|\frac{2k-1}{2k+2}x\right| = |x| < 1,$$

i.e., the r.o.c. is 1. The power series (1) only valid for $-1 < x < 1$ for real x.*

*The power series (1) also converges for x=1. To see this, let $c_k = a_k(-1)^{k-1}$, either by using Stirling's formula, or by noting

$$\begin{aligned} \lim_{k\to\infty} a_k &= \lim_{k\to\infty} \frac{(2k-3)!}{k!(k-2)!2^{2k-2}} = \lim_{k\to\infty} \frac{1\times 3\times ...\times(2k-3)}{2\times 4\times ...\times(2k)} \\ &\leq \lim_{k\to\infty} \frac{1}{2k}(\frac{2k-3}{2k-2})^{k-1} = \lim_{k\to\infty}\frac{1}{2k}((1-\frac{1}{2k-2})^{2k-2})^{1/2} = \lim_{k\to\infty}\frac{1}{2k}e^{-1/2} = 0 \end{aligned}$$

and a_k decreasing, since

$$\frac{a_{k+1}}{a_k} = \frac{(2k+1)!k!(k-1)!(2k-1)2^{2k-1}}{(2k-1)!(k+1)!k!(2k+1)2^{2k+1}} = \frac{k-1/2}{k+1} < 1.$$

By using alternating sequence test, (1) converges for x=1.

1.1.1 Investigating partial sums: Programming Task

Two assistant programs calculating $\sum_{k=0}^{N} c_k := S_N$ and error is listed on page 15, named **Q1(N)**, **Q1plot(N)**,.

See results in Tbl.1 and plots in Fig.1 for S_N and error under various N. Values are corrected up to computer precision (15 digits).

N	S_N	$S_N - \sqrt{2}$	$\left\|S_N - \sqrt{2}\right\|$
5	1.425781250000000	0.011567687626905	0.011567687626905
10	1.409931182861328	-0.004282379511767	0.004282379511767
49	1.414621586476354	4.080241032589083e-04	4.080241032589083e-04
100	1.414073047717717	-1.405146553785652e-04	1.405146553785652e-04
500	1.414200956186002	-1.260618709353345e-05	1.260618709353345e-05

Table 1: S_N, absolute error and its magnitude

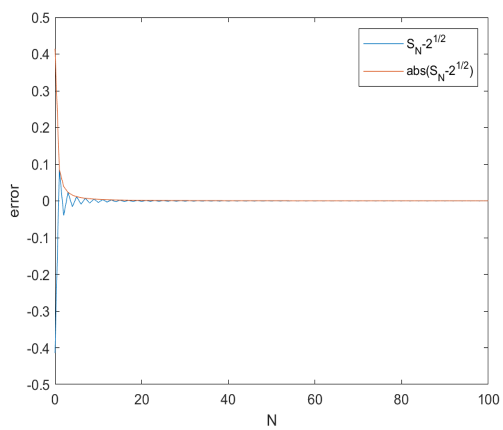

Figure 1: The evolution of error as N increases

As N increases, the absolute error oscillates around 0 with decreasing reduction in its magnitude. As N $\rightarrow \infty$, the error becomes 0.

1.2 Question 2

1.2.1 The Padé approximant $R_{L,L}$(x): Programming Task

A program determining the Padé approximant $R_{L,L}$(x) is listed on page 16, named **Q2(L,x)**. Two assistant programs are listed on page 16-19, named **Q2error(L,x)**, **Q2compare(L,x)**. Assume $R_{0,0} = 1$.

Take x=1. See results in Tbl.2 and plots in Fig.2 for $R_{L,L}$ and error under various L. Values are corrected up to computer precision (15 digits).

L	$R_{L,L}(1)$	$R_{L,L}(1) - \sqrt{2}$	$\lvert R_{L,L}(1) - \sqrt{2} \rvert$
1	1.400000000000000	-0.014213562373095	0.014213562373095
5	1.414213551646055	-0.000000010727040	0.000000010727040
8	1.414213562372821	-0.000000000000274	0.000000000000274
9	1.414213562373087	-0.000000000000008	0.000000000000008
10	1.414213562373095	-0.000000000000000	0.000000000000000
11	1.414213562373095	-0.000000000000000	0.000000000000000
16	1.414213562373095	0	0
35	1.414213562373093	-0.000000000000002	0.000000000000002
46	1.414213562373116	0.000000000000021	0.000000000000021

Table 2: $R_{L,L}(1)$, absolute error and its magnitude

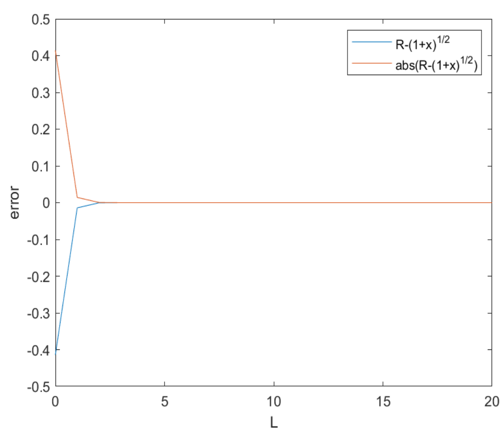

Figure 2: The evolution of error as L increases

As L increases, the absolute error keeps going up to 0 with decreasing increment and reaches exact 0 when L=16, i.e., the error is minimised. This is inherited from the fact that $\sqrt{2}$ is irrational hence cannot be represented by a fraction as $R_{L,L}(1)$ does. However, as L increases, the error can be reduced to zero since the series expansion of $f_1(x)$ converges at x=1. I.e., the **radius of convergence** of the power series (i.e., how close the true value can be written into a fraction under various L or the solution status of the simultaneous equations (4)(5) in the setup (e.g., whether singular or not)) determines the smallest value of the error. (As L continues to increase, the error becomes oscillating around 0 partially due to rounding errors).

The iterative improvement makes no difference. For Ax=b, assume A non-singular and start at y_0,

$$x \approx y_0 + \delta y_0 \Rightarrow A\delta y_0 \approx b - Ay_0 \Rightarrow \delta y_0 \approx A^{-1}b - y_0,$$
$$y_1 = y_0 + \delta y_0 \Rightarrow A\delta y_1 \approx b - A(y_0 + \delta y_0) = b - Ay_0 - A\delta y_0 \Rightarrow \delta y_1 \approx A^{-1}b - y_0 - \delta y_0.$$

I.e., δy_0, δy_1,... also converges to 0 with decreasing magnitude.

Compare estimates $R_{L,L}$ and S_N with N=L as L increases. See results in Fig.3.

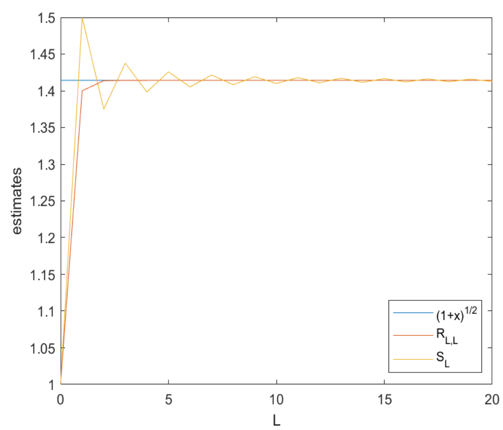

Figure 3: Comparison between $R_{L,L}$ and S_N as L increases

While $R_{L,L}$ goes up reaching $\sqrt{2}$ at a faster rate (at least at the start), S_N has steadily decreasing error (which is not the case for $R_{L,L}$ as shown in Tbl.2) with fluctuations. Hence the power series is more preferred to estimate $\sqrt{2}$ under specified accuracy requirements.

1.3 Question 3

1.3.1 Comparing power series estimates and Padé approximant estimates for f_1: Programming Task

Programs comparing the two estimates and calculating their errors are listed on page 19-22, named **Q3(L,a,x)**, **Q3error(L,x)**, **Q3errorii**. L is for both $R_{L,L}$ and S_N, a is an assistant value used to see how S_N behaves when expanded around x=a.

For L=20(=N), a=5†, x=10, results are shown in Fig.4-6.

It can be seen that when x(=10) is outside the r.o.c. $|x|<1$, as L increases the power series estimate S_N heavily fluctuates and is divergent (Fig.4-5) (despite its convergence when expanded around x=a=5), while the Padé approximant estimate still converges to $(1+x)^{1/2}$ very well (Fig.6).

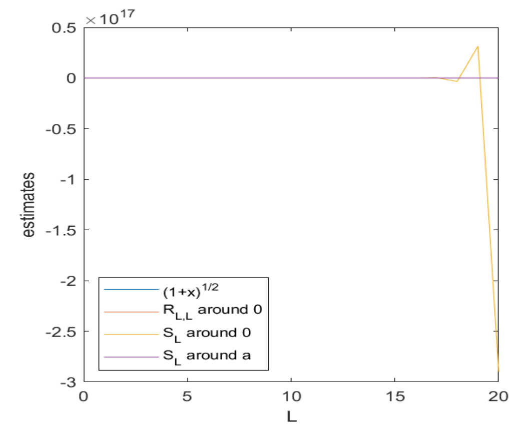

Figure 4: L=N=20, a=5, x=10

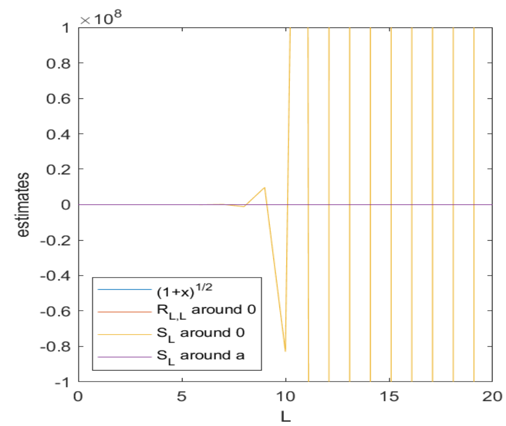

Figure 5: L=N=20, a=5, x=10(zoomed)

---

†Expand $(1+x)^{1/2}$ around x=a gives

$$(1+a)^{1/2}(1+\frac{x-a}{1+a})^{1/2} = \sum_{k=0}^{\infty}\binom{2k-1}{k}\frac{(1+a)^{1/2-k}(-1)^{k-1}}{(2k-1)2^{2k-1}}(x-a)^k.$$

Then

$$\lim_{k\to\infty}|\frac{c_{k+1}x^{k+1}}{c_k x^k}| = \lim_{k\to\infty}|\frac{(2k-1)(1+a)^{1/2-k-1}}{(2k+2)(1+a)^{1/2-k}}x| = \lim_{k\to\infty}|x(1+a)^{-1}| < 1,$$

i.e., $|x|<|1+a|$. Hence for a=5, r.o.c is 6, can choose x=10; for a=50, r.o.c is 51, can choose x=100.

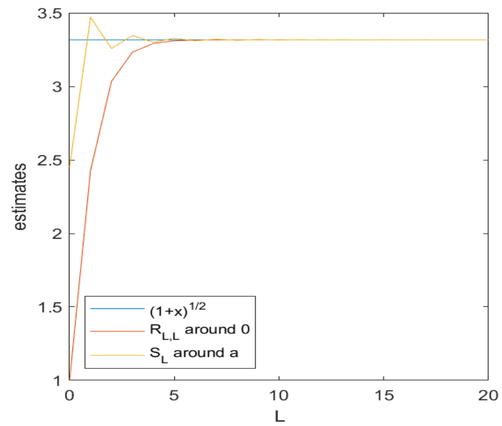

Figure 6: L=N=20, a=5, x=10

Changing x from 10 to 100, as L increases, the error in the Padé approximant estimates converges to 0 at a slower rate at the beginning (Fig.7). For x large (=100), however, unlike small x (=10), there are fluctuations and several large deviations from 0 as L continuous to grow (Fig.8). This **implies** that although the Padé approximant estimate can give a good approximation to the true value for small x, it fails to do so for x large, especially when the target polynomial has sufficiently large number of terms.

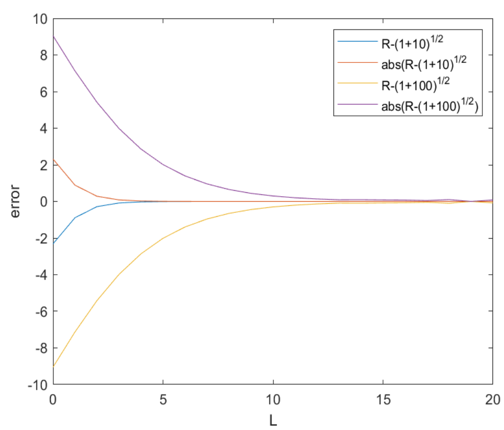

Figure 7: Compare errors at different x: L=N=20, a=5, x=10, 100

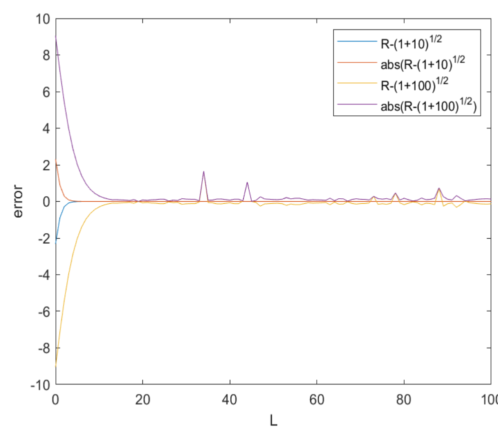

Figure 8: Compare errors at different x: L=N=100, a=5, x=10, 100

1.4 Question 4

1.4.1 Comparing power series estimates and Padé approximant estimates for f_2: Programming Task

Programs comparing the two estimates are listed on page 22-25, named **Q4(L,x)**, **Q4ii(L)**, **Q4iicompare**. L is for both $R_{L,L}$ and S_N.

Results in Fig.9-11.

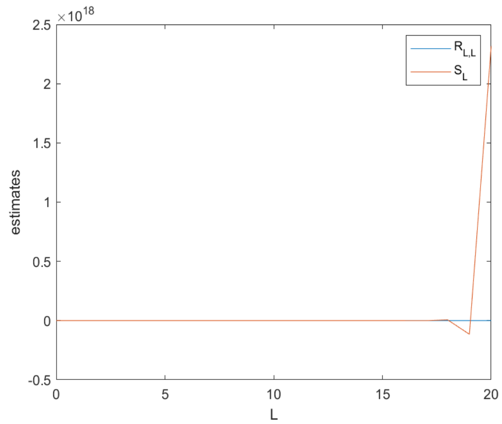

Figure 9: $R_{L,L}$, S_L against L for x=1

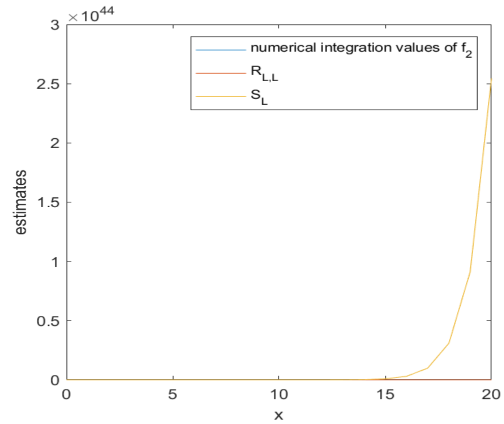

Figure 10: numerical integration values of f_2, $R_{20,20}$, S_{20} against x

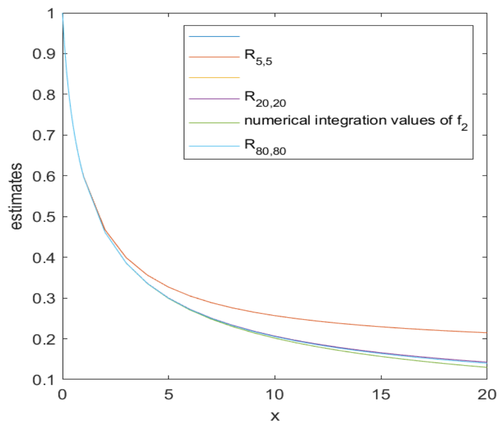

Figure 11: numerical integration values of f_2, $R_{L,L}$ against x for varies L

As shown in Fig.9-10, the power series estimate makes sense for small x and diverges when x is large as expected. Its error also fluctuates significantly when L is large, which is consistent with our previous discuss. In comparison, the Padé approximant estimate continues to give good approximations (despite fluctuations) to the values of $f_2(x)$ obtained from numerical integration. The gap variance in Fig.11 shows that its error is reduced as L increases.

1.5 Question 5

1.5.1 Investigating the poles and zeros of the Padé approximant: Programming Task

By writing simultaneous equations in matrix form, locate the poles and zeros of the Padé approximant of different functions via programs listed on page 25-29, named **Q5(L)** (for $f_1(x)$), **Q5s** (for $f_3(x)$ to $f_6(x)$)[‡], **Q5ii(L,M)**[§], **Q5_x(L,xmin,xmax,h)**, **Q5_xii**[¶].

Results in Fig.12-20.

[‡] To calculate coefficients of $f_5(x)$ and $f_6(x)$ (f_3, f_4 are trivial by taylor expansions), note

$$
\begin{aligned}
f_5(x) &= e^x(1+x)^{-1} = (1 + x + \frac{x^2}{2!} + ...)(1 - x + x^2 - ...) = \sum_{k=0}^{\infty}\left(\sum_{i=0}^{k}\frac{1}{i!}(-1)^{k-i}\right)x^k, \\
f_6(x) &= (1+x+x^2)^{1/2} \\
&= 1 + \sum_{k=1}^{\infty}\binom{2k-1}{k}\frac{(-1)^{k-1}}{(2k-1)2^{2k-1}}\,x^k(1+x)^k \\
&= 1 + \sum_{k=1}^{\infty}\sum_{i=0}^{k}\binom{2k-1}{k}\binom{k}{i}\frac{(-1)^{k-1}}{(2k-1)2^{2k-1}}\,x^{k+i} \\
&= 1 + \sum_{k=1}^{\infty}\sum_{r=k}^{2k}\binom{2k-1}{k}\binom{k}{r-k}\frac{(-1)^{k-1}}{(2k-1)2^{2k-1}}\,x^{r} \\
&= 1 + \sum_{r=1}^{\infty}\left(\sum_{k=\lceil r/2\rceil}^{r}\binom{2k-1}{k}\binom{k}{r-k}\frac{(-1)^{k-1}}{(2k-1)2^{2k-1}}\right)x^{r}.
\end{aligned}
$$

[§] An assistant program to check the results when need M=L+1.

[¶] Two assistant programs to plot $R_{L,L}$ against x.

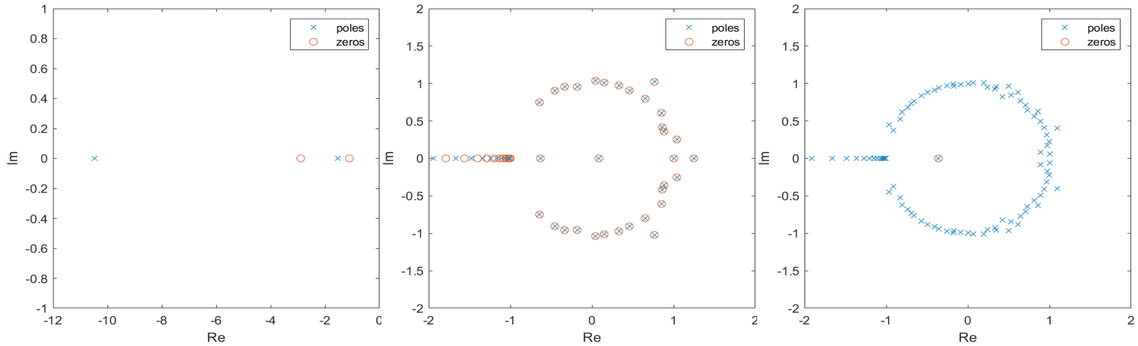

Figure 12: f_1 with L=2, 50 (zoomed), 100 (zoomed)

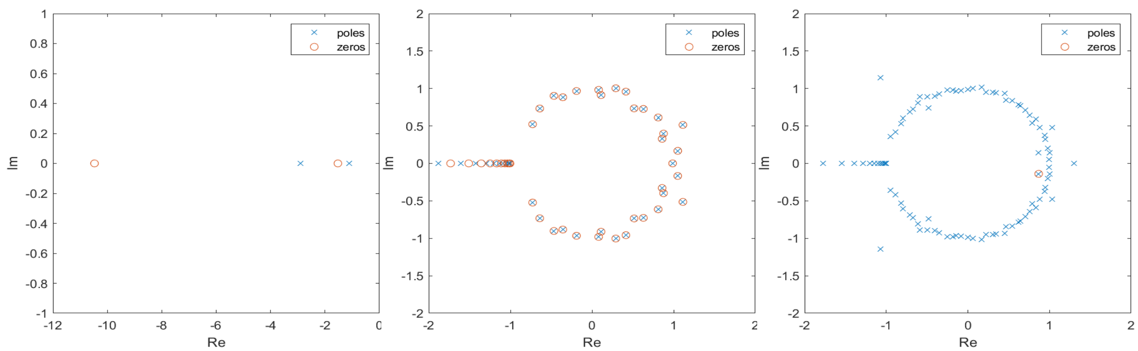

Figure 13: f_3 with L=2, 50 (zoomed), 100 (zoomed)

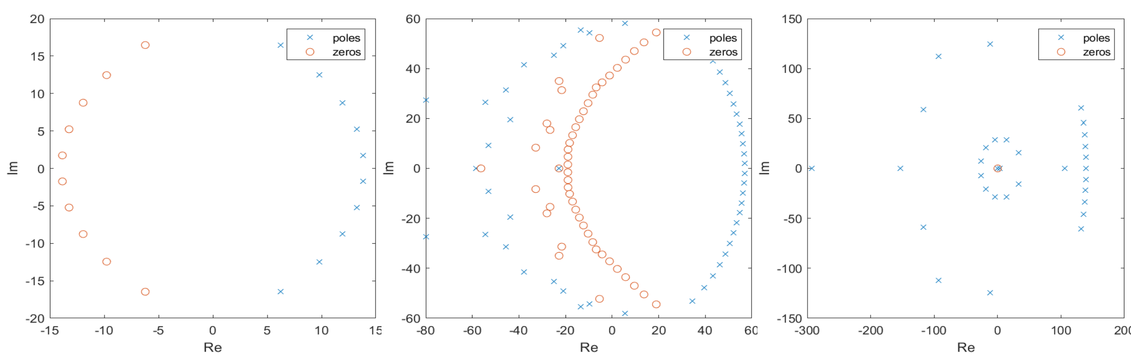

Figure 14: f_4 with L=10, 50, 150

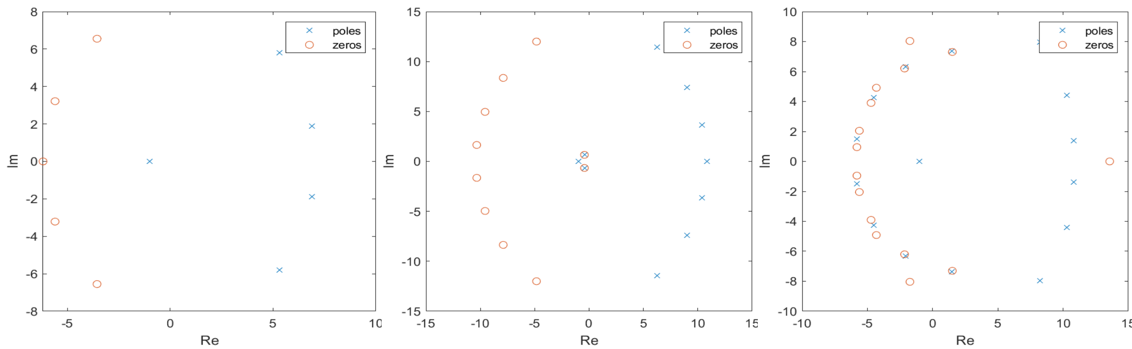

Figure 15: f_5 with L=5, 10, 15

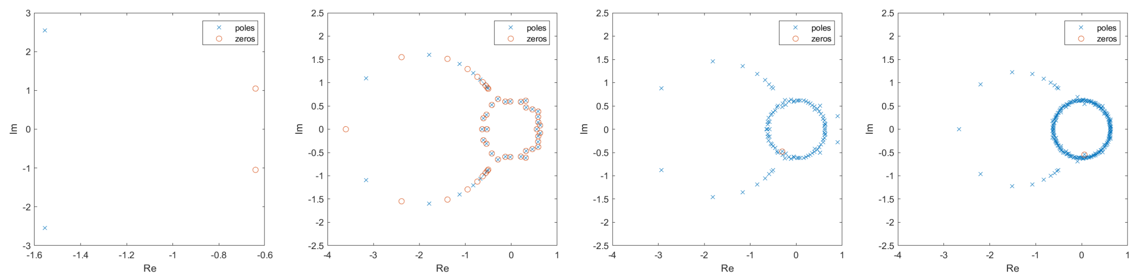

Figure 16: f_6 with L=2, 50 (zoomed), 100 (zoomed), 200 (zoomed)

	zero(s)	pole(s)	branch point(s)	branch cut(s)
f_1	-1	/	-1	$\{z \mid -\infty < z \leq -1\}$
f_3	/	-1	-1	$\{z \mid -\infty < z \leq -1\}$
f_4	/	/	/	/
f_5	/	-1	/	/
f_6	$-\frac{1}{2} \pm \frac{\sqrt{3}}{2}i$	/	$-\frac{1}{2} \pm \frac{\sqrt{3}}{2}i$	$\{z = re^{\pm 2\pi i/3} \mid 0 \leq r \leq 1\}$

Table 3: Certain features of the target functions

- f_1 – poles and zeros form an alternating sequence on the branch cut and accumulate at the branch point -1. When L is small, can see clearly from Fig.12(a) ($R_{2,2} = \frac{\frac{5x^2}{16}+\frac{5x}{4}+1}{\frac{x^2}{16}+\frac{3x}{4}+1}$ with simple poles at $-6 \pm 2\sqrt{5}$ and simple zeros at $-2 \pm \frac{2\sqrt{5}}{5}$) there is a zero at approximate -1, consistent with what we have found in Tbl.3 for f_1. When L increases, both poles and zeros appear more and more on the branch cut and also evenly on the unit circle (**cancelling** each other[‖]), forming an inverted 'C' shape, possibly a **natural boundary**[**], with its corners pointing towards the branch point (Fig.12(b)). As L $\to \infty$ (Fig.12(c)), only 1 zero left at -1[††] as indicated in Tbl.3.

- f_3 − similar as f_1 but with a pole instead of a zero at -1 (Fig.13) ($R_{2,2} = \frac{\frac{x^2}{16}+\frac{3x}{4}+1}{\frac{5x^2}{16}+\frac{5x}{4}+1}$ with simple poles at $-2 \pm \frac{2\sqrt{5}}{5}$ and simple zeros at $6 \pm 2\sqrt{5}$, exactly inverted compared to f_1). As L$\to\infty$, there is an unstable zero (i.e. changes places when varying L) on the unit circle (Fig.13(c)).

- f_4 − all zeros stay on the left-half plane, all poles stay on the right-half plane, each forming a semicircle (Fig.14(a)). When L increases, more of both accumulate on **larger** semicircles of their own with a small portion of them distributes on the left-half plane outside the semicircle[‡‡] (Fig.14(b)). As L $\to \infty$ (Fig.14(c)), both poles and zeros vanish , consistent with what we have found in Tbl.3 for f_4.

- f_5 − similar as f_4 but with a pole at -1 (Fig.15).

- f_6 − when L is small, can see clearly from Fig.16(a) ($R_{2,2} = \frac{\frac{53x^2}{80}+\frac{17x}{20}+1}{\frac{9x^2}{80}+\frac{7x}{20}+1}$ with simple poles at $-\frac{14}{9} \pm \frac{2\sqrt{131}}{9}i$ and simple zeros at $-\frac{34}{53} \pm \frac{2\sqrt{771}}{53}i$) there is two zeros near $-\frac{1}{2} \pm \frac{\sqrt{3}}{2}i$, consistent with what we have found in Tbl.3 for f_6. When L increases, poles and zeros form an alternating sequence on the large 'C' shape centred at approximate -2 with corners pointing towards the branch points $-\frac{1}{2} \pm \frac{\sqrt{3}}{2}i$. There are also poles and zeros cancelling each other on the small circle centred at 0, possibly a **natural boundary**. As L $\to \infty$, the large 'C' **shrinks** to the right − an alternative curved branch cut connected 2 branch points − and more poles accumulate evenly on the small circle while zeros vanish.

[‖]Possible computation error.

[**]Reference: [1]Hiroaki S. Yamada and Kensuke S. Ikeda, 'A Numerical Test of Pade Approximation for Some Functions with Singularity', 5-10(April 1, 2014), April 16, 2021. arXiv: 1308.4453v2 [math-ph] 28 Mar 2014.

[††]Approximate -1 by Fig.12(c) under finite L due to a compromise between positioning precision and computational accuracy.

[‡‡]Possible computation error due to insufficient numerical accuracy.

[*]Possible computation error as more terms appear in the Padé approximant.

[†]Possible computation error. Should have 2 left at the location where the branch points are, as indicated in Tbl.3.

Hence we can conclude, in terms of anomalous poles and zeros,

- f_1 − dense poles and zeros evenly on the unit circle generated by the Padé approximant not matched by the target function, possibly a **natural boundary**. Zeros not matched for small L but better matched for large L.

- f_3 − similar as f_1 with an extra unstable zero on the unit circle that is not matched by the target function (that only has a fixed pole at z=-1).

- f_4 − poles and zeros on the two semicircles (and some outside) not matched by the target function. As L→ ∞, zeros vanish while poles accumulate on large circle, indicating an **essential singularity** at $z = \infty$ (which agrees with the target function since $e^{1/z}$ has an essential singularity at z=0).

- f_5 − similar as f_4 with extra poles and zeros between semicircles.

- f_6 − poles and zeros on the 'C' shaped region (unevenly) and the small circle - possibly a **natural boundary** - (evenly) not matched by the target function. As L→ ∞, zeros vanish while poles accumulate more on the small circle, disagreeing with the target function. On the other hand, the 'C' shaped region consisting of poles shrinks and collapses to the branch cut connecting two branch points, which is consistent with the target function.

When estimating f_6 along the real x-axis (Fig.17),

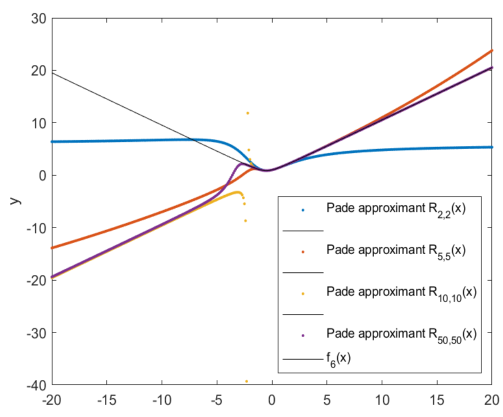

Figure 17: The Padé approximants compared to exact values of f_6

there are different levels of deviation from f_6 for large $|x|$ as L increases. For certain values of L, e.g. L=10, there is also a break point at approximate x=-2.3, where the

Padé approximants hit a pole and go to $\pm\infty$, leading to bad estimation. f_6 is best approximated in a small neighbourhood of x=0.

f_6 has two branch points on either side of the real x-axis, leading to possible discontinuity error when performing the Padé approximants in the case branch cut is chosen to connect two branch points.

There are also a large number of **noisy poles** (and zeros) around the natural boundary (which cuts the real x-axis) leading to confusion when doing the analysis using the Padé approximants (see Fig.16).

Computation error.

2 Programs

Note: Some programs listed on this pdf have $'return'$ added after excessively long texts, which needs to be removed before testing.

2.1 Question 1

2.1.1 Q1(N)

```
function Q1(N)
% e.g. Q1(10)
i=1; S=1;
while i<=N
    c=nchoosek(2*i-1,i)*(-1)^(i-1)/(2^(2*i-1)*(2*i-1));
    S=S+c; i=i+1;
end
fprintf( 'the partial sum of c_0, c_1,...,c_N is')
S
fprintf('(S_N-2^0.5) is')
S-2^0.5
fprintf('the absolute value of (S_N-2^0.5) is')
abs(S-2^0.5)
end
```

2.1.2 Q1plot(N)

```
function Q1plot(N)
% e.g. Q1plot(10)
i=1; S=zeros(1,N+1); S(1)=1; X=[0:N];
while i<=N
    c=nchoosek(2*i-1,i)*(-1)^(i-1)/(2^(2*i-1)*(2*i-1));
    S(i+1)=S(i)+c; i=i+1;
end
Y=S-2^0.5;Z=abs(S-2^0.5);
plot(X,Y,X,Z)
xlabel('N');ylabel('error')
legend('S_N-2^{1/2}','abs(S_N-2^{1/2})')
end
```

2.2 Question 2

2.2.1 Q2(L,x)

```
function Q2(L,x)
% e.g. Q2(10,1)
i=1; C=zeros(1,2*L+1); C(1)=1;
while i<=2*L
    C(i+1)=nchoosek(2*i-1,i)*(-1)^(i-1)/(2^(2*i-1)*(2*i-1));
    i=i+1;
end
i=1; A=zeros(L);
while i<=L
    A(1:L,i)=C(L+2-i:2*L+1-i).';
    i=i+1;
end
Q=mldivide(A,-C(L+2:2*L+1).');
i=1; B=zeros(L+1,L);
while i<=L
    B(i+1:L+1,i)=C(1:L+1-i);
    i=i+1;
end
P=(C(1:L+1).'+B*Q).';
X=zeros(1,L+1);
i=0;
while i<=L
    X(i+1)=x^i;
    i=i+1;
end
R=(P*X.')/(1+X(2:L+1)*Q);
fprintf('R_{L,L} is')
R
end
```

2.2.2 Q2error(L,x)

```
function Q2error(L,x)
% L - higher end of the range of L, R=1 when L=0.
```

```
% e.g. Q2error(20,1)
XX=0:L;R=zeros(1,L+1);R(1)=1;j=1;k=1;
while k<=L
% content
i=1; C=zeros(1,2*k+1); C(1)=1;
while i<=2*k
    C(i+1)=nchoosek(2*i-1,i)*(-1)^(i-1)/(2^(2*i-1)*(2*i-1));
    i=i+1;
end
i=1; A=zeros(k);
while i<=k
    A(1:k,i)=C(k+2-i:2*k+1-i).';
    i=i+1;
end
Q=mldivide(A,-C(k+2:2*k+1).');
i=1; B=zeros(k+1,k);
while i<=k
    B(i+1:k+1,i)=C(1:k+1-i);
    i=i+1;
end
P=(C(1:k+1).'+B*Q).';
X=zeros(1,k+1);
i=0;
while i<=k
    X(i+1)=x^i;
    i=i+1;
end
R(j+1)=(P*X.')/(1+X(2:k+1)*Q);
j=j+1; k=k+1;
end
% end of content
fprintf( 'R is up to suffix L')
R
fprintf( 'R-(1+x)^{1/2} is up to suffix L')
R-(1+x)^0.5
plot(XX,R-(1+x)^0.5,XX,abs(R-(1+x)^0.5));
xlabel('L');ylabel('error');
legend('R-(1+x)^{1/2}','abs(R-(1+x)^{1/2})')
```

```
Error_min=min(abs(R-(1+x)^0.5))
end
```

2.2.3 Q2compare(L,x)

```
function Q2compare(L,x)
% L for S_L and for R_{L,L}
% e.g. Q2compare(20,1)
XX=0:L;R=zeros(1,L+1);R(1)=1;S=R;j=1;k=1;
Y=zeros(1,L+1)+(1+x)^0.5;
i=1;
while i<=L
    c=nchoosek(2*i-1,i)*(-1)^(i-1)/(2^(2*i-1)*(2*i-1))*x^i;
    S(i+1)=S(i)+c;  i=i+1;
end
while k<=L
% content
i=1; C=zeros(1,2*k+1); C(1)=1;
while i<=2*k
    C(i+1)=nchoosek(2*i-1,i)*(-1)^(i-1)/(2^(2*i-1)*(2*i-1));
    i=i+1;
end
i=1; A=zeros(k);
while i<=k
    A(1:k,i)=C(k+2-i:2*k+1-i).';
    i=i+1;
end
Q=mldivide(A,-C(k+2:2*k+1).');
i=1; B=zeros(k+1,k);
while i<=k
    B(i+1:k+1,i)=C(1:k+1-i);
    i=i+1;
end
P=(C(1:k+1).'+B*Q).';
X=zeros(1,k+1);
i=0;
while i<=k
    X(i+1)=x^i;
```

```
    i=i+1;
end
R(j+1)=(P*X.')/(1+X(2:k+1)*Q);
j=j+1; k=k+1;
end
% end of content
plot(XX,Y,XX,R,XX,S)
xlabel('L');ylabel('estimates');
legend('(1+x)^{1/2}','R_{L,L}','S_L')
end
```

2.3 Question 3

2.3.1 Q3(L,a,x)

```
function Q3(L,a,x)
% f_1 , S_N expand around x=a, R_{L,L} only around 0
% L for S_L and for R_{L,L}
% e.g. Q3(20,5,10)
XX=0:L;R=zeros(1,L+1);R(1)=1;S1=R;S2=R;S2(1)=(1+a)^(1/2);j=1;k=1;
Y=zeros(1,L+1)+(1+x)^0.5;
i=1;
while i<=L
    c1=nchoosek(2*i-1,i)*(-1)^(i-1)/(2^(2*i-1)*(2*i-1))*x^i;
    c2=(1+a)^(1/2-i)*nchoosek(2*i-1,i)*(-1)^(i-1)/(2^(2*i-1)*(2*i-1))*
    (x-a)^i;
    S1(i+1)=S1(i)+c1; S2(i+1)=S2(i)+c2;
    i=i+1;
end
while k<=L
% content
i=1; C=zeros(1,2*k+1); C(1)=1;
while i<=2*k
    C(i+1)=nchoosek(2*i-1,i)*(-1)^(i-1)/(2^(2*i-1)*(2*i-1));
    i=i+1;
end
i=1; A=zeros(k);
while i<=k
```

```
    A(1:k,i)=C(k+2-i:2*k+1-i).';
    i=i+1;
end
Q=mldivide(A,-C(k+2:2*k+1).');
i=1; B=zeros(k+1,k);
while i<=k
    B(i+1:k+1,i)=C(1:k+1-i);
    i=i+1;
end
P=(C(1:k+1).'+B*Q).';
X=zeros(1,k+1);
i=0;
while i<=k
    X(i+1)=x^i;
    i=i+1;
end
R(j+1)=(P*X.')/(1+X(2:k+1)*Q);
j=j+1; k=k+1;
end
% end of content
subplot(1,3,1)
plot(XX,Y,XX,R,XX,S1,XX,S2)
xlabel('L');ylabel('estimates');
legend('(1+x)^{1/2}','R_{L,L} around 0','S_L around 0','S_L around a')
subplot(1,3,2)
plot(XX,Y,XX,R,XX,S1,XX,S2)
xlabel('L');ylabel('estimates');
legend('(1+x)^{1/2}','R_{L,L} around 0','S_L around 0','S_L around a')
axis([0,L,-10^8,10^8])
subplot(1,3,3)
plot(XX,Y,XX,R,XX,S2)
xlabel('L');ylabel('estimates');
legend('(1+x)^{1/2}','R_{L,L} around 0','S_L around a')
end
```

2.3.2 Q3error(L,x)

```
function Q3error(L,x)
```

```
% L - higher end of the range of L, R=1 when L=0.
% e.g. Q3error(20,10),Q3error(20,100)
XX=0:L;R=zeros(1,L+1);R(1)=1;j=1;k=1;
while k<=L
% content
i=1; C=zeros(1,2*k+1); C(1)=1;
while i<=2*k
    C(i+1)=nchoosek(2*i-1,i)*(-1)^(i-1)/(2^(2*i-1)*(2*i-1));
    i=i+1;
end
i=1; A=zeros(k);
while i<=k
    A(1:k,i)=C(k+2-i:2*k+1-i).';
    i=i+1;
end
Q=mldivide(A,-C(k+2:2*k+1).');
i=1; B=zeros(k+1,k);
while i<=k
    B(i+1:k+1,i)=C(1:k+1-i);
    i=i+1;
end
P=(C(1:k+1).'+B*Q).';
X=zeros(1,k+1);
i=0;
while i<=k
    X(i+1)=x^i;
    i=i+1;
end
R(j+1)=(P*X.')/(1+X(2:k+1)*Q);
j=j+1; k=k+1;
end
% end of content
plot(XX,R-(1+x)^0.5,XX,abs(R-(1+x)^0.5));
xlabel('L');ylabel('error');
end
```

2.3.3 Q3errorii

```
Q3error(20,10);hold on;
Q3error(20,100)
legend('R-(1+10)^{1/2}','abs(R-(1+10)^{1/2}','R-(1+100)^{1/2}',
'abs(R-(1+100)^{1/2})')

Q3error(100,10);hold on;
Q3error(100,100)
legend('R-(1+10)^{1/2}','abs(R-(1+10)^{1/2}','R-(1+100)^{1/2}',
'abs(R-(1+100)^{1/2})')
```

2.4 Question 4

2.4.1 Q4(L,x)

```
function Q4(L,x)
% L for S_L and for R_{L,L}
% e.g. Q4(20,1)
XX=0:L;R=zeros(1,L+1);R(1)=1;S=R;i=1;j=1;k=1;
while i<=L
    c=factorial(i)*(-x)^i;
    S(i+1)=S(i)+c; i=i+1;
end
while k<=L
% content
i=1; C=zeros(1,2*k+1); C(1)=1;
while i<=2*k
    C(i+1)=factorial(i)*(-1)^i;
    i=i+1;
end
i=1; A=zeros(k);
while i<=k
    A(1:k,i)=C(k+2-i:2*k+1-i).';
    i=i+1;
end
Q=mldivide(A,-C(k+2:2*k+1).');
i=1; B=zeros(k+1,k);
while i<=k
    B(i+1:k+1,i)=C(1:k+1-i);
```

```
    i=i+1;
end
P=(C(1:k+1).'+B*Q).';
X=zeros(1,k+1);
i=0;
while i<=k
    X(i+1)=x^i;
    i=i+1;
end
R(j+1)=(P*X.')/(1+X(2:k+1)*Q);
j=j+1; k=k+1;
end
% end of content
plot(XX,R,XX,S)
xlabel('L');ylabel('estimates');
legend('R_{L,L}','S_L')
end
```

2.4.2 Q4ii(L)

```
function Q4ii(L)
% L for S_L and for R_{L,L}
% x=[0:0.1:0.9,1:20]
% e.g. Q4ii(20)
XXX=[0:0.1:0.9,1:20];RR=zeros(1,30);SS=RR;p=1;
Y=[1,0.91563334,0.85211088,0.80118628,0.75881459,0.72265723,0.69122594,
0.66351027,0.63879110,0.61653779,0.59634736,0.46145532,0.38560201,
0.33522136,0.29866975,0.27063301,0.24828135,0.22994778,0.21457710,
0.20146425,0.19011779,0.18018332,0.17139800,0.16356229,0.15652164,
0.15015426,0.14436271,0.13906806,0.13420555,0.12972152];
while p<=30
R=zeros(1,L+1);R(1)=1;S=R;i=1;j=1;k=1;
while i<=L
    c=factorial(i)*(-XXX(p))^i;
    S(i+1)=S(i)+c;  i=i+1;
end
while k<=L
% content
```

```
i=1; C=zeros(1,2*k+1); C(1)=1;
while i<=2*k
     C(i+1)=factorial(i)*(-1)^i;
     i=i+1;
end
i=1; A=zeros(k);
while i<=k
     A(1:k,i)=C(k+2-i:2*k+1-i).';
     i=i+1;
end
Q=mldivide(A,-C(k+2:2*k+1).');
i=1; B=zeros(k+1,k);
while i<=k
     B(i+1:k+1,i)=C(1:k+1-i);
     i=i+1;
end
P=(C(1:k+1).'+B*Q).';
X=zeros(1,k+1);
i=0;
while i<=k
     X(i+1)=XXX(p)^i;
     i=i+1;
end
R(j+1)=(P*X.')/(1+X(2:k+1)*Q);
j=j+1; k=k+1;
end
SS(p)=S(L+1);RR(p)=R(L+1);
p=p+1;
end
% end of content
subplot(1,2,1)
plot(XXX,Y,XXX,RR,XXX,SS)
xlabel('x');ylabel('estimates');
legend('numerical integration values of f_{2}','R_{L,L}','S_L')
subplot(1,2,2)
plot(XXX,Y,XXX,RR)
xlabel('x');ylabel('estimates');
legend('numerical integration values of f_{2}','R_{L,L}')
```

```
end
```

2.4.3 Q4iicompare

```
%Q4 compare L=5, L=20 and L=80
Q4ii(5);hold on;Q4ii(20);hold on;Q4ii(80)
legend('','R_{5,5}','','R_{20,20}','numerical integration values of
f_{2}','R_{80,80}')
```

2.5 Question 5

2.5.1 Q5(L)

```
function Q5(L)
% e.g. Q5(10)
i=1; C=zeros(1,2*L+1); C(1)=1;
while i<=2*L
    C(i+1)=nchoosek(2*i-1,i)*(-1)^(i-1)/(2^(2*i-1)*(2*i-1));
    i=i+1;
end
i=1; A=zeros(L);
while i<=L
    A(1:L,i)=C(L+2-i:2*L+1-i).';
    i=i+1;
end
Q=mldivide(A,-C(L+2:2*L+1).');
i=1; B=zeros(L+1,L);
while i<=L
    B(i+1:L+1,i)=C(1:L+1-i);
    i=i+1;
end
P=(C(1:L+1).'+B*Q).';

syms x;k=1;f1=P(1);f2=1;
while k<=L
    f1=f1+P(k+1)*x^k;f2=f2+Q(k)*x^k;
    k=k+1;
end
```

```
R=f1/f2
S=poles(R),Z=solve(R==0,x),
Sr=real(S);Si=imag(S);Zr=real(Z);Zi=imag(Z);
subplot(1,2,1)
plot(Sr,Si,'x',Zr,Zi,'o')
legend('poles','zeros')
xlabel('Re');ylabel('Im');
subplot(1,2,2)
plot(Sr,Si,'x',Zr,Zi,'o')
legend('poles','zeros')
xlabel('Re');ylabel('Im');
axis([-2,2,-2,2])
end
```

2.5.2 Q5s

```
%Q5 programs branch substitution for line 5 in program Q5_1

%f_1
C(i+1)=nchoosek(2*i-1,i)*(-1)^(i-1)/(2^(2*i-1)*(2*i-1));

%f_3
C(i+1)=nchoosek(2*i-1,i)*(-1)^i/(2^(2*i-1));

%f_4
C(i+1)=1/factorial(i);

%f_5
j=0;
    while j<=i
        C(i+1)=C(i+1)+(-1)^(i-j)/factorial(j);
        j=j+1;
    end

%f_6
j=ceil(i/2);
    while j<=i
        C(i+1)=C(i+1)+nchoosek(2*j-1,j)*nchoosek(j,i-j)*(-1)^(j-1)/
```

```
          (2^(2*j-1)*(2*j-1));
          j=j+1;
    end
%f_6 change line 41 to
axis([-4,1,-2.5,2.5])
```

2.5.3 Q5ii(L,M)

```
function Q5ii(L,M)
% M=L+1
% e.g. Q5ii(10,11)
i=1; C=zeros(1,L+M+1); C(1)=1;
while i<=L+M
    j=0;
    while j<=i
         C(i+1)=C(i+1)+(-1)^(i-j)/factorial(j);
         j=j+1;
    end
    i=i+1;
end
i=1; A=zeros(M);
while i<=M
    A(1:M,i)=C(L+2-i:L+M+1-i).';
    i=i+1;
end
Q=mldivide(A,-C(L+2:L+M+1).');
i=1; B=zeros(L+1,L);
while i<=L
    B(i+1:L+1,i)=C(1:L+1-i);
    i=i+1;
end
P=(C(1:L+1).'+B*Q(1:L)).';

syms x;k=1;f1=P(1);f2=1;
while k<=L
    f1=f1+P(k+1)*x^k; f2=f2+Q(k)*x^k;
    k=k+1;
end
```

```
R=f1/f2;
S=poles(R);Z=solve(R==0,x);
Sr=real(S);Si=imag(S);Zr=real(Z);Zi=imag(Z);
subplot(1,2,1)
plot(Sr,Si,'x',Zr,Zi,'o')
legend('poles','zeros')
xlabel('Re');ylabel('Im');
subplot(1,2,2)
plot(Sr,Si,'x',Zr,Zi,'o')
legend('poles','zeros')
xlabel('Re');ylabel('Im');
axis([-2,2,-2,2])
end
```

2.5.4 **Q5**$_x(L, xmin, xmax, h)$

```
function Q5_x(L,xmin,xmax,h)
% e.g. Q5_x(10,-2,2,0.1)
i=1; C=zeros(1,2*L+1); C(1)=1;
while i<=2*L
    j=ceil(i/2);
    while j<=i
        C(i+1)=C(i+1)+nchoosek(2*j-1,j)*nchoosek(j,i-j)*(-1)^(j-1)/
        (2^(2*j-1)*(2*j-1));
        j=j+1;
    end
    i=i+1;
end
i=1; A=zeros(L);
while i<=L
    A(1:L,i)=C(L+2-i:2*L+1-i).';
    i=i+1;
end
Q=mldivide(A,-C(L+2:2*L+1).');
i=1; B=zeros(L+1,L);
while i<=L
    B(i+1:L+1,i)=C(1:L+1-i);
    i=i+1;
```

```
end
P=(C(1:L+1).'+B*Q).';

k=1;f1=P(1);f2=1;X=xmin:h:xmax;
while k<=L
    f1=f1+P(k+1)*X.^k;f2=f2+Q(k)*X.^k;
    k=k+1;
end
R=f1./f2;
Y=(1+X+X.^2).^(1/2);
plot(X,R,'.',X,Y,'k');
xlabel('x');ylabel('y');
legend('Pade approximant R(x)','f_6(x)');
end
```

2.5.5 $Q5_x ii$

```
%Q5_xii
Q5_x(2,-20,20,0.1);hold on;
Q5_x(5,-20,20,0.1);hold on;
Q5_x(10,-20,20,0.1);hold on;
Q5_x(50,-20,20,0.1)
legend('Pade approximant R_{2,2}(x)','','Pade approximant R_{5,5}(x)',
'','Pade approximant R_{10,10}(x)','','Pade approximant R_{50,50}(x)',
'f_6(x)')
```