1

Given the generating function $e^{2xt-t^2} = \sum_{n=0}^{\infty} H_n(x) t^n / n!$.

a. Derive the Rodrigues’ formula.

b. Directly from the generating function, show that $H'_n(x) = 2n H_{n-1}(x)$.

---

a. First we complete the square:

\[ e^{2xt-t^2} = e^{-t^2+2xt-x^2+x^2} = e^{x^2} e^{-(t-x)^2}. \]

We know

\[ H_n(x) = \left. \frac{\partial^n}{\partial t^n} \left( e^{2xt-t^2} \right) \right|_{t=0} = \left. e^{x^2} \frac{\partial^n}{\partial t^n} \left( e^{-(t-x)^2} \right) \right|_{t=0}. \]

Making the change of variables $y = t - x$ and then $z = -y$, we have

\[ H_n(x) = \left. e^{x^2} \frac{\partial^n}{\partial y^n} e^{-(t-x)^2} \right|_{t=0} = \left. e^{x^2} \frac{\partial^n}{\partial y^n} e^{-y^2} \right|_{y=-x} = (-1)^n e^{x^2} \frac{\partial^n}{\partial x^n} e^{-x^2}. \]

b. We can use equality of mixed partials to show

\[ H'_n(x) = \left. \frac{\partial}{\partial x} \frac{\partial^n}{\partial t^n} e^{2xt-t^2} \right|_{t=0} = \left. \frac{\partial^n}{\partial t^n} \frac{\partial}{\partial x} e^{2xt-t^2} \right|_{t=0} \]

\[ = \left. \frac{\partial^n}{\partial t^n} 2te^{2xt-t^2} \right|_{t=0} = 2 \left. \frac{\partial^n}{\partial t^n} \left( \sum_{k=0}^{\infty} H_k(x) \frac{t^k}{k!} \right) \right|_{t=0} \]

\[ = 2 \sum_{k=n-1}^{\infty} H_k(x) \frac{(k+1)!}{(k+1-n)!} \frac{t^{k+1-n}}{k!} \left. \right|_{t=0} = 2H_{n-1}(x) \frac{n-1+1}{1} \]

\[ = 2n H_{n-1}(x). \]
a. Use G-S to orthogonalize the functions $1, x, x^2$ with respect to the IP

$$\langle f, g \rangle = \int_0^1 f(x)g(x)dx.$$ 

b. Show that we can generate (unnormalized) orthogonal polynomials through the choices

$$\alpha_k = \frac{\langle xp_k, p_k \rangle}{\langle p_k, p_k \rangle}, \quad \beta_k = \frac{\langle p_k, p_k \rangle}{\langle p_{k-1}, p_{k-1} \rangle},$$

in the recursion

$$p_{k+1} = (x - \alpha_k)p_k - \beta_k p_{k-1}.$$ 

c. Use the three term recursion formula (with $p_{-1} = 0$ and $p_0 = 1$), to obtain the same result as in part a.

---

a. Let $\phi_0 = 1$, $\phi_1 = x$, and $\phi_2 = x^2$. Then

$$\psi_0 = \phi_0 = 1$$
$$\psi_1 = \phi_1 - \frac{\langle \phi_1, \psi_0 \rangle}{\langle \psi_0, \psi_0 \rangle} \psi_0 = x - \frac{1}{2} \cdot \frac{1}{1} = x - \frac{1}{2}$$
$$\psi_2 = \phi_2 - \frac{\langle \phi_2, \psi_1 \rangle}{\langle \psi_1, \psi_1 \rangle} \psi_1 = \frac{x^2 - 1/12}{1/12} \cdot \frac{1}{x - 1/2} = x^2 - x + \frac{1}{6}.$$

b. Define $\alpha_k$ and $\beta_k$ appropriately. Set

$$p_{k+1} = (x - \alpha_k)p_k - \beta_k p_{k-1}.$$ 

Since $p_{k+1}$ is defined via $p_k$ and $p_{k-1}$ (which are assumed to be orthogonal to all previous $p_j$, $j \leq k$; think induction on $\mathbb{N}$), $p_{k+1}$ must be orthogonal to $p_j$, for $j \leq k - 2$. Therefore, it is sufficient to check that $p_{k+1}$ is orthogonal to $p_k$ and $p_{k-1}$.

$$\langle p_{k+1}, p_k \rangle = \langle (x - \alpha_k)p_k, p_k \rangle + 0 = \langle xp_k, p_k \rangle - \alpha_k \langle p_k, p_k \rangle = 0$$
$$\langle p_{k+1}, p_{k-1} \rangle = \langle xp_{k-1}, p_{k-1} \rangle + \langle -\beta_k p_{k-1}, p_{k-1} \rangle = \langle p_k, xp_{k-1} \rangle - \langle p_k, p_k \rangle$$
$$= \sum_{n=0}^k a_n p_n \cdot (x - \alpha_k) = a_k \langle p_k, p_k \rangle - \langle p_k, p_k \rangle = 0,$$

since we know $a_k = 1$ comes from Gram-Schmidt (we take $x^k$ and subtract lower order terms; thus, the highest degree monomial has coefficient 1). Therefore, this recursion relation generates orthogonal polynomials w.r.t. $\langle \cdot, \cdot \rangle$. 

---

2
c. Set \( p_{-1} = 0 \) and \( p_0 = 1 \). A simple computation gives \( \alpha_0 = 1/2 \). Then

\[
p_1 = (x - 1/2)1 - 0 = x - \frac{1}{2}.
\]

Using this,

\[
\alpha_1 = \frac{\int_0^1 x(x - 1/2)^2 dx}{\int_0^1 (x - 1/2)^2 dx} = \frac{1/24}{1/12} = \frac{1}{2}, \quad \beta_1 = \frac{1}{12} = \frac{1}{12}.
\]

Then

\[
p_2 = (x - 1/2)(x - 1/2) - 1/12 = x^2 - x + \frac{1}{6}.
\]

3

Start with the nowhere convergent Taylor series for the Stieltjes function. Use the code `t2cf` to generate a sequence of leading CF coefficients. Identify the CF coefficient pattern that emerges, and then use this pattern to evaluate \( f(2) \) as accurately as you can. Compare against an accurate result for \( f(2) \).

---

Proceeding formally, we have

\[
f(z) = \int_0^\infty \sum_{k=0}^\infty (-z^k e^{-t}) dt = \sum_{k=0}^\infty (-1)^k z^k \int_0^\infty t^k e^{-t} dt = \sum_{k=0}^\infty (-1)^k k! z^k.
\]

Using the provided code `t2cf`, we have the CF expansion

\[
f(z) = \frac{1}{1 + \frac{1}{z} + \frac{2}{z} + \frac{3}{z} + \frac{4}{z} + \ldots}
\]

We have the coefficients

\[c_1 = 1, \quad c_{2k} = c_{2k+1} = k.\]

Using the provided code `cf2y` and \( k = 10 \), I found

\[f(2) \approx 0.4619604954576314.\]

I then used Mathematica to find

\[f(2) \approx 0.4614553162418652.\]

So, I needed to beef up \( k \). Using \( k = 147 \) (the highest I could get without the code returning `NaN`), I have

\[f(2) \approx 0.4614553162418714.\]
Assuming the value coming out of Mathematica is correct, the CF approximation with \( k = 147 \) (which means 295 coefficients) is correct to 13 digits.

4

Convert the given Taylor expansion to its CF form and attempt to spot a closed form expansion for its CF coefficients. Make a plot over \( x \in [-5, 5], y \in [-5, 5] \) that includes the direct evaluation of the Taylor series, a high-order CF version, and the function \( f(x) = 1 + (1 + \cot(x))^{-1} \).

Using the provided code t2cf, we have the following CF expansion:

\[
f(x) = \frac{1}{1 + \frac{x/6}{1 + \frac{-2x/5}{1 + \frac{x/14}{1 + \frac{-2x/9}{1 + \vdots}}}}}
\]

This appears to follow the pattern

\[
c_0 = -c_1 = 1, c_2 = 2, \quad \text{and for } k \geq 2, \quad \begin{cases} 
  c_{2k} = -c_{2k+1} = \frac{1}{2(2k - 1)} & \text{if } k \text{ even} \\
  -c_{2k} = c_{2k+1} = \frac{2}{2k - 1} & \text{if } k \text{ odd.}
\end{cases}
\]

Below I’ve plotted the Taylor series, CF expansion with \( k = 10 \), and \( f(x) \) on the same set of axes. Note that the CF expansion is so good that we can’t see it under the plot of \( f(x) \). I approximated the max-norm error of the CF expansion: \( \|CF[f] - f\|_\infty \approx 1.65 \times 10^{-1} \). So, the CF expansion, even with \( k = 10 \), captures a lot about \( f \), including the asymptotes. This is quite remarkable, and it seems like voodoo witchcraft to me.

On the other side of things, the Taylor expansion is pretty awful. Over the interval \([-5, 5]\), the max-norm error is approximately \( \|T[f] - f\|_\infty \sim 10^7 \). This is quite awful, not that we really expected anything better.
Listing 1: GNU Octave to estimate \( f(2) \)

```octave
% GNU Octave
function [] = p3(N=10)

k = 0:1:20;
ak = (-1).^k.*factorial(k);
cfk = t2cf(ak);
cfn = ones([1 2*2+1]);
for n=2:N
    cfn(2*n:2*n+1) = [n n];
end
fprintf(1,'CF[f(2)] = %.16f\n',cf2y(2,cfn));
end
```

Listing 2: GNU Octave to plot \( f(x) \) and its Taylor and CF expansions

```octave
% GNU Octave
function [] = p4(N=10)
ak = [1 1 -1 4/3 -5/3 32/15 -122/45 1088/315 -277/63 15872/2835 -101042/14175];
cfk = t2cf(ak);
```
cfn = zeros([1 2*N+1]);
cfn(1:3) = [1 -1 2];
for n=2:N;
    if mod(n,2) == 0
        cfn(2*n) = 1/(2*(2*n-1));
        cfn(2*n+1) = -1/(2*(2*n-1));
    else
        cfn(2*n) = -2/(2*n-1);
        cfn(2*n+1) = 2/(2*n-1);
    end
end

x = -5:.05:5;
Tx = polyval(fliplr(ak), x);
CFx = cf2y(x,cfn);
fx = 1+1./(1+cot(x));
norm(Tx-fx,'inf')
norm(CFx-fx,'inf')
plot(x,Tx,x,CFx,x,fx);
axis([-5 5 -5 5]);
xlabel('x')
ylabel('y')
title('$T[x]$','$CF[f](x)$, and $f(x)$');
print('figures/p4.tikz','-dtikz','-S640x480');
end