Final project bootstrapping

Cheng-Feng-Yin

b09505009

December 2022

1 Find the function $\langle r^a \rangle (E)$:

Using the recurrence relation

$$\langle r^{a-1} \rangle = \frac{1}{8mE} \left[(1-a)(a(a-2) - 4l(l+1)) \langle r^{a-3} \rangle - 4(2a-1) \langle r^{a-2} \rangle \right]$$

$\langle r^{-1} \rangle = -2E$

$\langle r^0 \rangle = 1$

$\langle r \rangle = -\dfrac{2El(l+1)+3}{4E}$

$\langle r^2 \rangle = \dfrac{6El^2+6El-2E+5}{8E^2}$

$\langle r^3 \rangle = -\dfrac{70El(l+1)+18E(l(l+1)-2)(2El(l+1)+3)+14E(4l(l+1)-3)+105}{192E^3}$

$\langle r^4 \rangle = \dfrac{60E^2l^4+120E^2l^3-140E^2l^2-200E^2l+48E^2+140El^2+140El^2+140El-210E+63}{128E^4}$

$\langle r^5 \rangle = \dfrac{-40E^3l^6-120E^3l^5+200E^3l^4+600E^3l^3-160E^3l^2-480E^3l-420E^2l^4-840E^2l^3+1680E^2l^2+2100E^2l}{512E^5}$

$\dfrac{-1176E^2-630El^2-630El+1470E-231}{512E^5}$

$\langle r^6 \rangle = \dfrac{280E^3l^6+840E^3l^5-2240E^3l^4-5880E^3l^3+3976E^3l^2+7056E^3l-1440E^3+1260E^2l^4+2520E^2l^3}{1024E^6}$

$\dfrac{-7560E^2l^2-8820E^2l+8484E^2+1386El^2+1386El-4620E+429}{1024E^6}$

$\langle r^7 \rangle = \dfrac{-560E^4l^8-2240E^4l^7+7840E^4l^6+31360E^4l^5-27440E^4l^4-109760E^4l^3+20160E^4l^2+80640E^4l-10080E^3l^6}{16384E^7}$

$\dfrac{-30240E^3l^5+115920E^3l^4+282240E^3l^2-328944E^3l^2-475104E^3l+219168E^3-27720E^2l^4-55400E^2l^3+231000E^2l^2}{16384E^7}$

$\dfrac{+258720E^2l-374220E^2-24024El^2-24024El+108108E-6435}{16384E^7}$

$$\langle r^8\rangle = \frac{5040E^4l^8+20160E^4l^7-97440E^4l^6-362880E^4l^5+526512E^4l^4+1681344E^4l^3-848832E^4l^2-1753344E^4l+322560E^4}{32768E^8}$$
$$\frac{+36960E^3l^6+110880E^3l^5-572880E^3l^4-1330560E^3l^3+2310000E^3l^2+2993760E^3l-2314400E^3+72072E^2l^4+144144E^2l^3}{32768E^8}$$
$$\frac{-792792E^2l^2-864864E^2l+1741740E^2+51480El^2+51480El-300300E+12155}{32768E^8}$$
$$\langle r^9\rangle = \frac{-2016E^5l^{10}-10080E^5l^9+60480E^5l^8+302400E^5l^7-550368E^5l^6-2751840E^5l^5+1653120E^5l^4+8265600E^5l^3}{131072E^9}$$
$$\frac{-1161216E^5l^2-5806080E^5l-55440E^4l^8-221760E^4l^7+1404480E^4l^6+4989600E^4l^5-10515120E^4l^4-29604960E^4l^3}{131072E^9}$$
$$\frac{+25988160E^4l^2+41659200E^4l-17005824E^4-240240E^3l^6-720720E^3l^5+4804800E^3l^4+10810800E^3l^3-25825800E^3l^2}{131072E^9}$$
$$\frac{-31351320E^3l+35738560E^3-360360E^2l^4-720720E^2l^3+5045040E^2l^2+5405400E^2l-14402388E^2-218790El^2-218790El}{131072E^9}$$
$$\frac{+1604460E-46189}{131072E^9}$$

the code is as following:
where $y[k]$ is the $\langle r^{k-1}\rangle$, using the python sympy library to calculate symbols operations.

```
import sympy as smp
import numpy as np
import matplotlib.pyplot as plt
x=smp.symbols('x')
E=smp.symbols('E')
l=smp.symbols('l')
N=12
i=2
rn_2=1
rn_3=-2*E
y=[rn_3, rn_2]
for i in range(2,N):
  x=1/(8*E*i)*((1-i)*(i*(i-2)-4*l*(l+1))*rn_3-4*(2*i-1)*rn_2)
  y.append(x)
  rn_3=rn_2
  rn_2=x
  i=i+1
smp.simplify(y[10])
```

Figure 1: calculating $\langle r^a\rangle$

2 Shifted Hankel matrix:

The nth level shifted Hankel matrix is as follow: $\begin{pmatrix} \langle r \rangle & \langle r^2 \rangle & \langle r^3 \rangle & ... \\ \langle r^2 \rangle & \langle r^3 \rangle & \langle r^4 \rangle & ... \\ \langle r^3 \rangle & \langle r^4 \rangle & \langle r^5 \rangle & .... \\ ... & ... & ... & ... \end{pmatrix}$, i.e. $[H_s]_{ij} = \langle r^{i+j+1} \rangle_{0 \leq i,j \leq n}$

consider an operator

$$O = \sqrt{r} \sum_{i=0}^{n} c_i r^i$$

, when it acts on any state $|\alpha\rangle$,it's norm should be positive, hence

$$\langle \alpha | O^\dagger O | \alpha \rangle = \langle \alpha | \sum_{i,j} c_i^* c_j r^{i+j+1} | \alpha \rangle = \sum_{i,j} c_i^* c_j [H_s]_{ij} \geq 0$$

this proves that shifted Hankel matrix is positive semi-definite.

3 bootstrapping the hydrogen atom:

Using python code and scipy library, we have the following pattern.
The program first define a range of energies from -0.15 to near 0 with 100 points, then we define the recursion

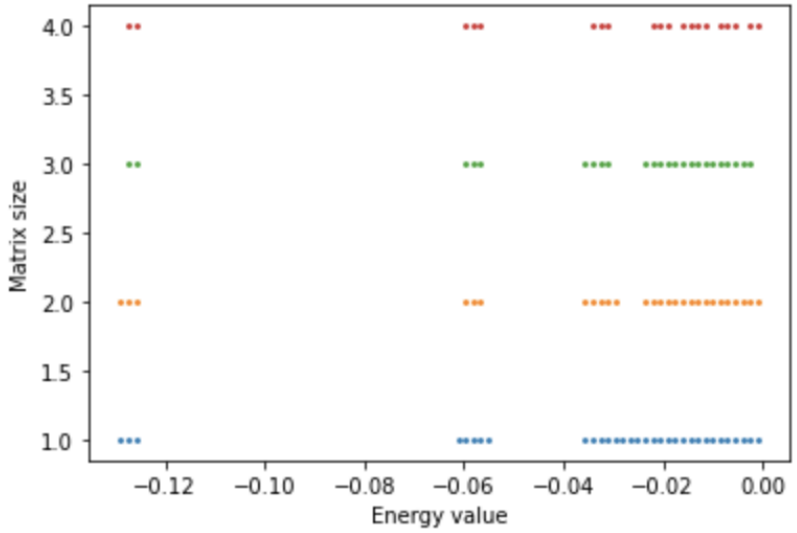

Figure 2: from size 12×12 to 15×15, with l=1

relation of $\langle r^a \rangle$, and Hankel matrix as $Hankel(n, l, E)$, the for loop selects the allowed value of E_r and put it into E_sel, then we plot the allowed energy with vertical axis the size of matrix. Since the program may overflow, we use the np.linalg.slogdet instead of original np.linalg.det, also we do only the n=12 to n=15 spectrum since the program will overflow for larger sizes of Hankel matrix. From the figure we see that there are energy eigenvalues at around -0.125, and at around -0.06, also at around -0.03, -0.02 and three

inside region $-0.02 \sim 0.00$. Which corresponds correctly with the spectrum of the exact hydrogen atom, $E_2 = -0.125$, $E_3 = -0.056$, $E_4 = -0.031$, $E_5 = -0.02$, $E_6 = -0.014$

```
import sympy as smp
import numpy as np
import matplotlib.pyplot as plt
from scipy.linalg import hankel
N,l=14,1
E_r=[]
E_sel=[]
y=[]
E_r=np.linspace(-0.15, -0.001,100)
num=len(E_r)
def r(n,l,E):
  if n==-1: return -2*E
  if n==0: return 1
  else: return 1./(8.*E*(n+1))*((-n)*((n+1)*(n-1)-4*l*(l+1))*r(n-2,l,E)-4*(2*n+1)*r(n-1,l,E))
def Hankel(n,l,E):
  rk=[]
  for i in range(n+1):
    if i==0:
      rk.append(1)
    else:
      rk.append(r(i,l,E))
  return hankel(rk)[0:int(n/2)+1,0:int(n/2)+1]
for n in range(11,N+1):
  E=[]
  for j in range(num):
    M=Hankel(2*n,l,E_r[j])
    sgn, logdet=np.linalg.slogdet(M)
    if sgn>=0: E.append(E_r[j])
  E_sel.append(E)
x_val=[]
for i in range(N-10):
  x_val.append(np.full(len(E_sel[i]),i+1))
for i in range(N-10):
  plt.scatter(E_sel[i], x_val[i], 3)
plt.xlabel("Energy value")
plt.ylabel("Matrix size")
```

Figure 3: code for hydrogen atom

4 bootstrapping modified spectrum:

Theoretically speaking we can look at the r^{-2} term, if $\exists k$ such that $l(l+1)+4 = k(k+1)$, where l is the angular momentum quantum number of the modified Hamiltonian, then it matches the state labeled by angular momentum quantum number k of the hydrogen atom. $(k-l)(k+l+1) = 4$ only gives us the $l = 1$ value corresponding to hydrogen atom of $k = 2$, hence we plot the following $l = 1$ from n=11 to n=15.

We modify the recursion relation of $\langle r^a \rangle$ as function $r(n, l, E)$:

$$\langle r^{a-1} \rangle = \frac{1}{8aE}[(1-a)(a(a-2) - 4l(l+1) - 16)\langle r^{a-3} \rangle - 4(2a-1)\langle r^{a-2} \rangle]$$

Note that some spectrum disappear at bigger size matrix, this is because there is a precision limit of the 100 dots, if the precise value of energy is within the width of a single dot, it will disappear, hence we can just look at the trend of the figure and see that there are two energy eigenvalues at region $-0.02 \sim 0.00$ and one at about 0.02, also one between $-0.04 \sim -0.02$ and about two between $-0.06 \sim -0.04$ etc.The others especially for the energy lower than -0.06 are not obvious.

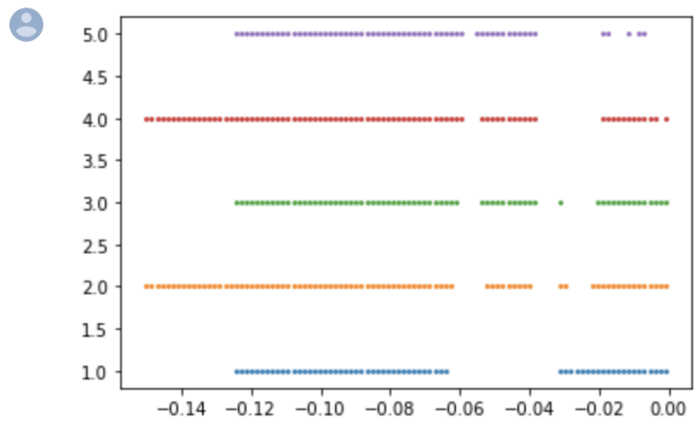

Figure 4: the dots are constrained energies and the labels 1 to 5 are the size of Hankel matrix from 11×11 to 15×15, with l=1

```
import sympy as smp
import numpy as np
import matplotlib.pyplot as plt
from scipy.linalg import hankel
N,l=14,1
E_r=[]
E_sel=[]
y=[]
E_r=np.linspace(-0.15, -0.001,100)
num=len(E_r)
def r(n,l,E):
  if n==-1: return -2*E
  if n==0: return 1
  else: return 1./8./(n+1)/E*((-n)*((n+1)*(n-1)-4*l*(l+1)-16)*r(n-2,l,E)-4*(2*n+1)*r(n-1,l,E))
def Hankel(n,l,E):
  rk=[]
  for i in range(n+1):
    if i==0:
      rk.append(1)
    else:
      rk.append(r(i,l,E))
  return hankel(rk)[0:int(n/2)+1,0:int(n/2)+1]
for n in range(10,N+1):
  E=[]
  for j in range(num):
    M=Hankel(2*n,l,E_r[j])
    sgn, logdet=np.linalg.slogdet(M)
    if sgn>=0: E.append(E_r[j])
  E_sel.append(E)
x_val=[]
for i in range(N-9):
  x_val.append(np.full(len(E_sel[i]),i+1))
for i in range(N-9):
  plt.scatter(E_sel[i], x_val[i], 3)
```

Figure 5: code for modified hydrogen

References

[1] Bootstrapping Simple QM Systems, David Berenstein, George Hulsey