

Exercise 1.1 solution

Example of a Schmidt decomposition

Contents

- [parameters:](#)
- [first get the ground state of the Heisenberg model:](#)
- [now reshape ground state vector into matrix:](#)
- [compute entanglement entropy](#)
- [And now the same for a random state](#)
- [And now plot distribution:](#)
- [Now part 4](#)

parameters:

```
L = 16;
Lhalf = L/2;
```

first get the ground state of the Heisenberg model:

```
H2 = getHeisenberg(1);
HL = getHL(H2,L);
% compute ground state:
[V,e0] = eigs(HL,1,'SA');
```

now reshape ground state vector into matrix:

(a left side and a right side of equal size):

```
MV = reshape(V,[2^Lhalf 2^Lhalf]);
[U,s,V] = svd(MV,'econ');
```

compute entanglement entropy

```
p = diag(s).*diag(s);
p = p/sum(p); % normalize
ee = - sum(p.*log(p));
disp0('Ground state energy:',e0, ', entanglement entropy:',ee);
```

```
Ground state energy:-6.9117, entanglement entropy:0.59231
```

And now the same for a random state

```
rand('seed',1);
vr = rand(2^L,1);
vr = vr / sqrt(vr'*vr); % normalize
```

```

vrenergy = vr'*HL*vr; % compute <vr|H|vr>
Mvr = reshape(vr,[2^Lhalf 2^Lhalf]);
[Ur,sr,Vr] = svd(Mvr,'econ');
% compute entanglement entropy
pr = diag(sr).*diag(sr);
pr = pr/sum(pr); % normalize
eer = - sum(pr.*log(pr));
disp0('Random state energy:',vrenergy, ', entanglement entropy:',eer);

```

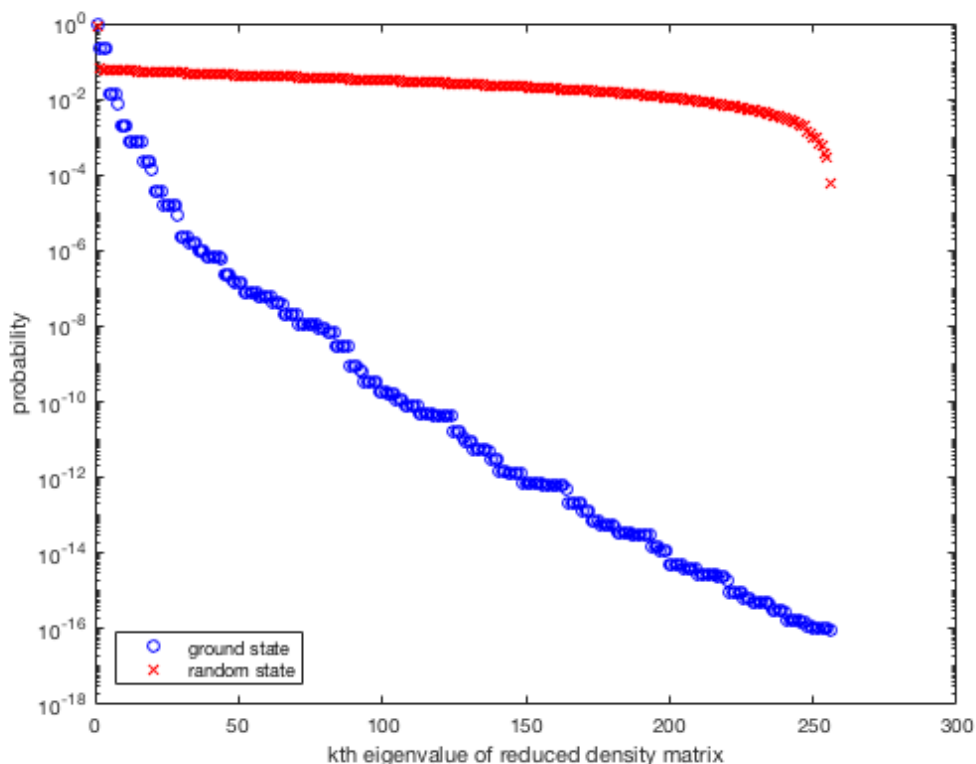
Random state energy:2.8149, entanglement entropy:1.8064

And now plot distribution:

```

figure;
set(gca,'FontSize',15);
semilogy(diag(s),'bo');
hold on;
semilogy(diag(sr),'rx');
xlabel('kth eigenvalue of reduced density matrix')
ylabel('probability');
le=legend('ground state','random state');
set(le,'Location','SouthWest');

```



Now part 4

plot the relative error of the energy of the two states as a function of the number of singular values (states) D kept.

```

vD = [2 4 6 8 10 20:20:100];

```

```

% store results in:
energies = zeros(numel(vD),1);
energiesr = zeros(numel(vD),1);

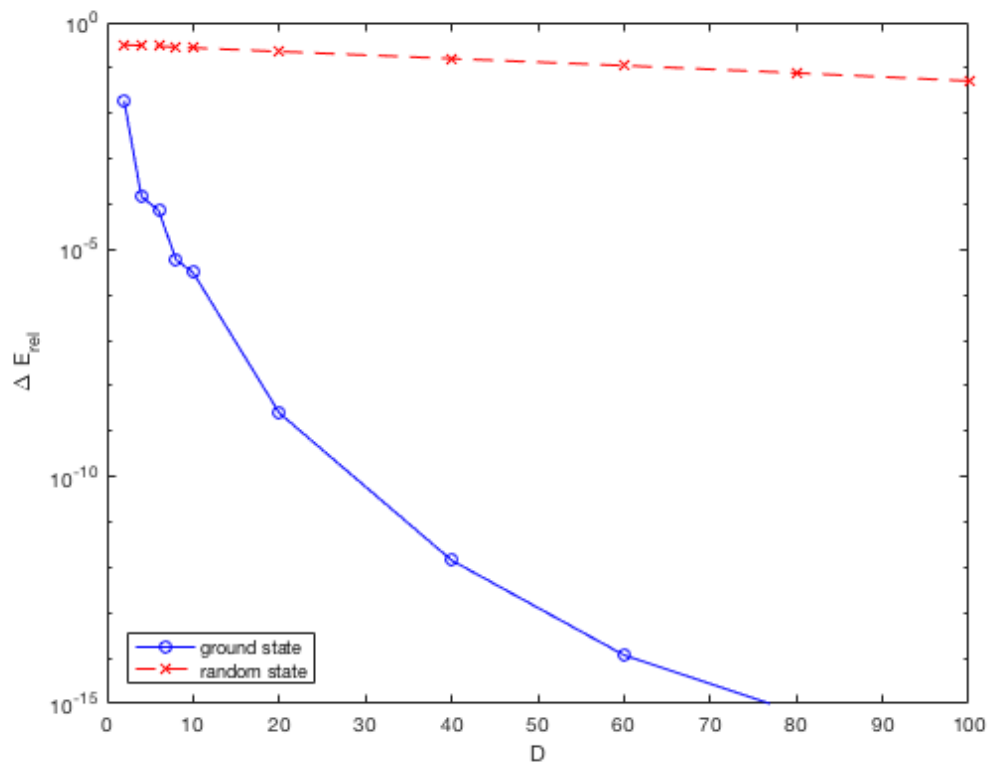
for k=1:numel(vD)
    D = vD(k); %current number of states to keep
    % compute truncated ground state
    truncstate = U(:,1:D) * s(1:D,1:D) * V(:,1:D)';
    truncstate = reshape(truncstate,[2^L,1]);
    % compute energy of truncated state
    energies(k) = truncstate' * HL * truncstate / (truncstate' * truncstate);

    % compute truncated random state
    truncstate = Ur(:,1:D) * sr(1:D,1:D) * Vr(:,1:D)';
    truncstate = reshape(truncstate,[2^L,1]);
    % compute energy of truncated state
    energiesr(k) = truncstate' * HL * truncstate / (truncstate' * truncstate);

end

figure;
set(gca, 'FontSize', 15);
semilogy(vD, (energies-e0)/abs(e0), 'bo-');
hold on;
semilogy(vD, (energiesr-vrenergy)/abs(vrenergy), 'rx--');
ylim([1e-15 1]);
xlabel('D');
ylabel('\Delta E_{rel}')
le=legend('ground state', 'random state');
set(le, 'Location', 'SouthWest');

```



Published with MATLAB® R2017a