Training tensor networks to approximate the distribution given by a dataset

Simple notebook demonstrating how to train a tensor network to learn the distribution of a dataset.

First import the dependencies and a tensor network model.

In [1]:

```
import numpy as np
import pickle
from tensornetworks.PositiveMPS import PositiveMPS
from tensornetworks.RealBorn import RealBorn
from tensornetworks.ComplexBorn import ComplexBorn
from tensornetworks.RealLPS import RealLPS
from tensornetworks.ComplexLPS import ComplexLPS
from tensornetworks.ComplexCircularLPS  import ComplexCircularLPS
```

Now load a dataset. A dataset needs to be an integer numpy array, with each row corresponding to a training example and each column to a categorical variable taking values in 0 to d-1.

In [2]:

```
with open('datasets/lymphography', 'rb') as f:
        a=pickle.load(f)
X=a[0]
X=X.astype(int)
```

Create a tensor network model (here we use a real `PositiveMPS`) with bond dimension `3` . Specify a learning rate, size of batch and number of epochs of training. Use 'verbose=True' to print the details during training.

In [3]:

```
mps =PositiveMPS(D=3, learning_rate=0.2, batch_size=20, n_iter=30, verbose=True)
```

Fit the model to the data

In [4]:
```
mps.fit(X)
```

```
Iteration 1, likelihood = 34.646, time = 0.01s
Iteration 2, likelihood = 29.435, time = 0.02s
Iteration 3, likelihood = 26.067, time = 0.02s
Iteration 4, likelihood = 23.803, time = 0.03s
Iteration 5, likelihood = 22.111, time = 0.03s
Iteration 6, likelihood = 20.868, time = 0.03s
Iteration 7, likelihood = 19.869, time = 0.03s
Iteration 8, likelihood = 19.098, time = 0.02s
Iteration 9, likelihood = 18.505, time = 0.02s
Iteration 10, likelihood = 18.010, time = 0.04s
Iteration 11, likelihood = 17.645, time = 0.02s
Iteration 12, likelihood = 17.284, time = 0.03s
Iteration 13, likelihood = 16.981, time = 0.02s
Iteration 14, likelihood = 16.748, time = 0.02s
Iteration 15, likelihood = 16.569, time = 0.02s
Iteration 16, likelihood = 16.419, time = 0.03s
Iteration 17, likelihood = 16.275, time = 0.02s
Iteration 18, likelihood = 16.179, time = 0.01s
Iteration 19, likelihood = 16.061, time = 0.03s
Iteration 20, likelihood = 15.945, time = 0.06s
Iteration 21, likelihood = 15.864, time = 0.03s
Iteration 22, likelihood = 15.791, time = 0.05s
Iteration 23, likelihood = 15.731, time = 0.07s
Iteration 24, likelihood = 15.652, time = 0.04s
Iteration 25, likelihood = 15.593, time = 0.02s
Iteration 26, likelihood = 15.556, time = 0.02s
Iteration 27, likelihood = 15.501, time = 0.04s
Iteration 28, likelihood = 15.479, time = 0.02s
Iteration 29, likelihood = 15.474, time = 0.04s
Iteration 30, likelihood = 15.418, time = 0.02s
```

Out[4]:
```
<tensornetworks.PositiveMPS.PositiveMPS instance at 0x0000000008C3C1C8>
```

Finally evaluate the negative log-likelihood of the fitted model.

In [5]:
```
mps.likelihood(X)
```

Out[5]:
```
15.418100458279229
```

Now create a tensor network model which is a complex LPS with bond dimension 3 and purification dimension of 2.

In [6]:
```
mps2 = ComplexLPS(D=3, learning_rate=0.2, batch_size=20, n_iter=30, verbose=True,
```

Fit the model to the data

In [7]:

```
mps2.fit(X)
```

```
Iteration 1, likelihood = 37.460, time = 0.07s
Iteration 2, likelihood = 35.159, time = 0.15s
Iteration 3, likelihood = 33.108, time = 0.09s
Iteration 4, likelihood = 31.391, time = 0.15s
Iteration 5, likelihood = 29.859, time = 0.09s
Iteration 6, likelihood = 28.465, time = 0.09s
Iteration 7, likelihood = 27.365, time = 0.13s
Iteration 8, likelihood = 26.370, time = 0.11s
Iteration 9, likelihood = 25.445, time = 0.12s
Iteration 10, likelihood = 24.647, time = 0.11s
Iteration 11, likelihood = 23.915, time = 0.13s
Iteration 12, likelihood = 23.246, time = 0.09s
Iteration 13, likelihood = 22.683, time = 0.09s
Iteration 14, likelihood = 22.122, time = 0.10s
Iteration 15, likelihood = 21.598, time = 0.10s
Iteration 16, likelihood = 21.152, time = 0.09s
Iteration 17, likelihood = 20.741, time = 0.09s
Iteration 18, likelihood = 20.354, time = 0.10s
Iteration 19, likelihood = 20.019, time = 0.09s
Iteration 20, likelihood = 19.704, time = 0.10s
Iteration 21, likelihood = 19.429, time = 0.09s
Iteration 22, likelihood = 19.169, time = 0.14s
Iteration 23, likelihood = 18.931, time = 0.09s
Iteration 24, likelihood = 18.706, time = 0.09s
Iteration 25, likelihood = 18.502, time = 0.09s
Iteration 26, likelihood = 18.314, time = 0.09s
Iteration 27, likelihood = 18.137, time = 0.09s
Iteration 28, likelihood = 17.989, time = 0.12s
Iteration 29, likelihood = 17.834, time = 0.09s
Iteration 30, likelihood = 17.695, time = 0.12s
```

Out[7]:

```
<tensornetworks.ComplexLPS.ComplexLPS instance at 0x0000000008C35108>
```

Finally evaluate the negative log-likelihood of the fitted model.

In [8]:

```
mps2.likelihood(X)
```

Out[8]:

```
17.69471330584164
```

Now create a tensor network model which is a complex circular LPS with bond dimension 3 and purification dimension of 2.

In [9]:

```
mps3 =ComplexCircularLPS(D=3, learning_rate=0.2, batch_size=20, n_iter=30, verbose=True,
```

Fit the model to the data

In [10]:

```
mps3.fit(X)
```

```
Iteration 1, likelihood = 37.460, time = 0.07s
Iteration 2, likelihood = 35.159, time = 0.15s
Iteration 3, likelihood = 33.108, time = 0.09s
Iteration 4, likelihood = 31.391, time = 0.15s
Iteration 5, likelihood = 29.859, time = 0.09s
Iteration 6, likelihood = 28.465, time = 0.09s
Iteration 7, likelihood = 27.365, time = 0.13s
Iteration 8, likelihood = 26.370, time = 0.11s
Iteration 9, likelihood = 25.445, time = 0.12s
Iteration 10, likelihood = 24.647, time = 0.11s
Iteration 11, likelihood = 23.915, time = 0.13s
Iteration 12, likelihood = 23.246, time = 0.09s
Iteration 13, likelihood = 22.683, time = 0.09s
Iteration 14, likelihood = 22.122, time = 0.10s
Iteration 15, likelihood = 21.598, time = 0.10s
Iteration 16, likelihood = 21.152, time = 0.09s
Iteration 17, likelihood = 20.741, time = 0.09s
Iteration 18, likelihood = 20.354, time = 0.10s
Iteration 19, likelihood = 20.019, time = 0.09s
Iteration 20, likelihood = 19.704, time = 0.10s
Iteration 21, likelihood = 19.429, time = 0.09s
Iteration 22, likelihood = 19.169, time = 0.14s
Iteration 23, likelihood = 18.931, time = 0.09s
Iteration 24, likelihood = 18.706, time = 0.09s
Iteration 25, likelihood = 18.502, time = 0.09s
Iteration 26, likelihood = 18.314, time = 0.09s
Iteration 27, likelihood = 18.137, time = 0.09s
Iteration 28, likelihood = 17.989, time = 0.12s
Iteration 29, likelihood = 17.834, time = 0.09s
Iteration 30, likelihood = 17.695, time = 0.12s
```

Out[10]:

```
<tensornetworks.ComplexCircularLPS.ComplexCircularLPS instance at 0x0000000008C35108>
```

Finally evaluate the negative log-likelihood of the fitted model.

In [11]:

```
mps3.likelihood(X)
```

Out[11]:

```
15.860130584515034
```