HOMEWORK 7

YangGao
9083410275

Instructions: There is no need to submit the latex source or any code. You can choose any programming language, as long as you implement the algorithm from scratch. Use this latex file as a template to develop your homework. Submit your homework on time as a single pdf file to Canvas. Please check Piazza for updates about the homework.

1 Chow-Liu Algorithm [100 pts]

Suppose we wish to construct a directed graphical model for 3 features X, Y, and Z using the Chow-Liu algorithm. We are given data from 100 independent experiments where each feature is binary and takes value T or F. Below is a table summarizing the observations of the experiment:

X	Y	Z	Count
T	T	T	36
T	T	F	4
T	F	T	2
T	F	F	8
F	T	T	9
F	T	F	1
F	F	T	8
F	F	F	32

1. Compute the mutual information $I(X, Y)$ based on the frequencies observed in the data. (20 pts)
2. Compute the mutual information $I(X, Z)$ based on the frequencies observed in the data. (20 pts)
3. Compute the mutual information $I(Z, Y)$ based on the frequencies observed in the data. (20 pts)
4. Which undirected edges will be selected by the Chow-Liu algorithm as the maximum spanning tree? (20 pts)
5. Root your tree at node X, assign directions to the selected edges. (20 pts)

2 Solutions

Solution 1.1

$I(X, Y) = 0.278$

```
MI = 0
for x in [True, False]:
  df_x = df[df['X']==x]
  for y in [True, False]:
    df_y = df[df['Y']==y]
    df_xy = df_x[df_x['Y']==y]
    MI += sum(df_xy['Count']/total) * np.log2(sum(df_xy['Count'])/total/(sum(df_x['Count'])/total * sum(df_y['Count'])/total))
print(MI)
```

```
0.27807190511263774
```

Solution 1.2

$I(X, Y) = 0.133$

```
MI = 0
for x in [True, False]:
  df_x = df[df['X']==x]
  for z in [True, False]:
    df_z = df[df['Z']==z]
    df_xz = df_x[df_x['Z']==z]
    MI += sum(df_xz['Count'])/total * np.log2(sum(df_xz['Count'])/total/(sum(df_x['Count'])/total * sum(df_z['Count'])/total))
print(MI)
```

```
0.13284496180903213
```

Solution 1.3

$I(X, Y) = 0.397$

```
MI = 0
for y in [True, False]:
  df_y = df[df['Y']==y]
  for z in [True, False]:
    df_z = df[df['Z']==z]
    df_yz = df_y[df_y['Z']==z]
    MI += sum(df_yz['Count'])/total * np.log2(sum(df_yz['Count'])/total/(sum(df_y['Count'])/total * sum(df_z['Count'])/total))
print(MI)
```

```
0.39731260974948646
```

Solution 1.4

The selected undirected edges are: X-Y, Z-Y, as these have the highest edge weights

Solution 1.5

X→Y, Y→Z