Tutorial 5

Gustav Kjellberg

February 2020

1 Task 0

We can illustrate the difference as two graphs with nodes Grade (G), Contentment level (C) and Disclose (D). The first graph would then look like this: $(G) \rightarrow (D)$, which is D-connected. The second model's graph would then look like $(G) \rightarrow \boldsymbol{(C)} \rightarrow (D)$ where $\boldsymbol{(C)}$ is observed, thus fulfilling the condition for D-separation, i.e. $(G) \perp (D)|(C)$.

2 Task 1

2.1 Seed = 1337

Learnt parameters

---

p(x) = [0.568 0.432]
p(y) = [0.294 0.706]
p(z—x=0, y=0) = [0.24137931 0.75862069]
p(z—x=0, y=1) = [0.63451777 0.36548223]
p(z—x=1, y=0) = [0.88333333 0.11666667]
p(z—x=1, y=1) = [0.1025641 0.8974359]

True parameters

---

p(x) = [0.6 0.4]
p(y) = [0.3 0.7]
p(z—x=0, y=0) = [0.2 0.8]
p(z—x=0, y=1) = [0.7 0.3]
p(z—x=1, y=0) = [0.9 0.1]
p(z—x=1, y=1) = [0.1 0.9]

2.2 Seed = 2018

Learnt parameters

---

p(x) = [0.586 0.414]
p(y) = [0.316 0.684]
p(z—x=0, y=0) = [0.20212766 0.79787234]
p(z—x=0, y=1) = [0.70351759 0.29648241]
p(z—x=1, y=0) = [0.890625 0.109375]
p(z—x=1, y=1) = [0.09090909 0.90909091]

3 Task 2

3.1 Seed = 1337

When we have missing values, we'll have to estimate the value using the joint, for example if x is missing we get the following

$$P(X|y,z) = \frac{P(X,y,z)}{P(z,y)} = \frac{P(X)P(z|X,y)}{\Sigma_x P(z|X,y)} \tag{1}$$

Learnt parameters

---

p(x) = [0.57321943 0.42678057]
p(y) = [0.28296361 0.71703639]
p(z—x=0, y=0) = [0.31614406 0.68385594]
p(z—x=0, y=1) = [0.5655918 0.4344082]
p(z—x=1, y=0) = [0.7679099 0.2320901]
p(z—x=1, y=1) = [0.18363023 0.81636977]

3.2 Seed = 2018

Learnt parameters

---

p(x) = [0.61139578 0.38860422]
p(y) = [0.31516116 0.68483884]
p(z—x=0, y=0) = [0.29649342 0.70350658]
p(z—x=0, y=1) = [0.64311197 0.35688803]
p(z—x=1, y=0) = [0.73331858 0.26668142]
p(z—x=1, y=1) = [0.18802168 0.81197832]

4 Task 3

Altering the generating values to be initialized as $[0.01, 0.99]$ for both px and py we obtain some interesting results when having 100 data points.
p(x) = [6.68785304e-151 1.00000000e+000]

p(y) = [0.02924309 0.97075691]

As we can see above, $\Sigma p(x) > 1$, though incredibly close to 1.0.

When increasing the number of data points by a factor 10, we obtain much better results, i.e
p(x) = [0.01044059 0.98955941]
p(y) = [0.00849308 0.99150692]

Further, having increased the number of data points to 10000 and the number of iterations to 1000 we obtain results very close to the true parameters:
Learnt parameters

---

p(x) = [0.59639318 0.40360682]
p(y) = [0.30089667 0.69910333]

Setting the probability for one outcome to 0 we obtain an assertion error as that would potentially lead to that we divide by zero.

5 Task 4

The definition of MAR says that the model is MAR if, for all $x_{hidden} \in val(X_{hidden})$, we have that o_x is independent of x_{hidden} given x_{obs} [Assignment 5, page 4]. This no longer hold, as if $o_x = x_{obs}$ then we know that $o_y = y_{hidden}$ and the other way around.

Without having run the code, I assume that our learnt parameters will be much more far from the true values as we now have 50% missing values for the set $X \cap Y$.

I can not tell a significant difference between the learnt parameters and the learnt marginals, but it is noticeable that the estimations requires more computation time as we need to marginalize either $X||Y$ for each observation. My intuition tells me that the marginals should have been better than the learnt parameters as that is what we "learn" for each observation as we have to marginalize out either $X||Y$, i.e for an observation x_{obs} we would get:

$$p(z|x, Y) = \frac{p(x, Y, z)}{p(x, Y)} = \Sigma_y p(z|x, Y) \tag{2}$$