Tutorial 2

Gustav Kjellberg

January 2020

1 Task 1

1.1 Task 1.1

The group with the smallest representation would be $age(> 23)$ as the value is 0. From Lecture 1, we know that the probability of a Naive Bayes Net is represented by

$$P(X_1, X_2, X_3, ..., X_N, C) = \prod_{i=1...N} P(X_i|Pa(X_i)) = P(C) \prod_{i=1...N} P(X_i|C) \quad (1)$$

Thus, if we then have

$$P(Age = (> 23)|Delay = (\geq (2)) = 0 \quad (2)$$

we will end up with a zero product. This could be handled by additive smoothing, i.e Laplace Smoothing, adding a pseudo-count.

1.2 Task 1.2

Yes, they are the same. As *delay* has no parents, the Conditional Probability Distribution (CPD) becomes the marginal distribution $P(delay)$.

2 Task 2

2.1 Task 2.1

Given a prior $age =<= 20$, we can define the query as:

```
q = ve.query(variables = ['delay'],evidence = {'age': '<=20'})
```

This results in a table where

$$\phi(delay = 0) = 0.8010 \quad (3)$$

Here we look at the probability of *delay* having a certain value having observed $age = '<= 20'$, whereas in 1.1 we were looking at $P(age|delay)$, i.e the conditional probability of *age* for any given *delay*.

2.2 Task 2.2

The age group with the lowest probability is $P(age = (> 23)|delay = (0)) = 0.0441$ whilst the group with the highest is $P(age = (<= 20)|delay = (0)) = 0.7696$.

2.3 Task 2.3

The values are the same as in the relative frequencies, as we assume that the features of the Naive Bayes Net are conditionally independent given the class, i.e $X_i \perp X_{-i}|C$ where X are all features, X_{-i} are all features besides X_i and C is the class or root.

Thus, we should theoretically be able to use the relative frequency to approximate $P(age = (x)|delay = (y))$ where y is the observed value of *delay* and x is the outcome we are want to know the probability for, by the following, where $\mathbf{D}$ is the data:

$$P(age = (y)|delay = (0)) = \frac{|(\mathbf{D}(age = (y) \cap delay = (0)))|}{|(\mathbf{D}(delay = 0))|} \tag{4}$$

as *age* is conditionally independent of all other features. [2]

2.4 Task 2.4

Using the following query

```
q = ve.map_query(variables = ['age'],evidence = {'delay': '0'})
```

we obtain the result $\{'age' :'<= 20'\}$, i.e the same as in section *2.2*.

3 Task 3

3.1 Task 3.1

See code

3.2 Task 3.2

The CPD will cover all possible combinations, as $age, gender, avg\_cs, avg\_mat \perp delay$ does not hold in this setting. The number of possible combinations, i.e

the number of CPDs will be:

$$|\overline{P(delay|age, gender, avg\_cs, avg\_mat)}| = |\overline{age}| * |\overline{gender}| * |\overline{avg\_cs}| * |\overline{avg\_mat}| * |\overline{delay}| = 3 * 2 * 4 * 4 * 4 = 384 \tag{5}$$

3.3 Task 3.3

The size of the data for *delay* is 265, i.e fewer samples than we have CPDs, thus there should be cases in which we have no data for the specific combination. In the cases where we have no data, the probabilities for those occurrences are thus 0. The reasoning to why that is similar to that in section *1.*

3.4 3.4

The CPD for the observations for which we have no data, should be 0. The reason behind this is as follows, assume that we have a combination of features for which we have no data, and let's denote these as $(a, g, cs, mat)_{na} = a, g, cs, mat \notin \mathbf{D}$

$$\begin{aligned} P(delay|(a, g, cs, mat)_{na}) &= \frac{|\overline{\mathbf{D}}(delay \cap (a, g, cs, mat)_{na})|}{|\overline{\mathbf{D}}(a, g, cs, mat)|} \\ &= \frac{0}{|\overline{\mathbf{D}}(a, g, cs, mat)|} = 0 \end{aligned} \tag{6}$$

3.5 3.5

The biggest error is yields from when $delay = 1$, for which we obtains a relative error $\epsilon = 0.0112$

As *delay* is not independent of all factors [*age, gender, avg_mat, avg_cs*] we can not expect to the relative frequencies to be equivalent to the marginal distribution of *delay.* Though, the factors themselves are independent and thus, their relative frequencies should correspond to their CPDs.

4 Task 4

Kullback-Leibler(KL) divergence is used when we approximate an unknown distribution $P(\mathbf{x})$ with $Q(\mathbf{x})$ to tell us how much information we are missing by approximating, i.e it gives us the amount of additional information required to specify a value of $\mathbf{x}$ by using $Q(\mathbf{x})$.The KL-divergence does not satisfy the needs to be a matric as is is not symmetrical between distributions, that being $KL(P||Q) \not\equiv KL(Q||P)$.[1, p.55]

4.1 Task 4.1

Given the equation for KL divergence below

$$KL(P||Q) = -\int P(\mathbf{x}) ln\{\frac{Q(\mathbf{x})}{P(\mathbf{x})}\} \tag{7}$$

[1, p.55] we see that the entropy $-> \infty$ as $x-> 0$

The reason to why this only effects the model with reversed edges, i.e the logistic regression model is because model 1, the Bayes Net, has marginal distributions equal to the relative frequencies, thus the fraction becomes 1.

4.2 Task 4.3

The lines of code do the following in respected order

1. amount of query results in which the there was 2 evidence parameters.
2. amount of query results in which the there was 2 evidence parameters and Naive Bayes net yields a smaller relative entropy than the model with reversed edges.
3. The opposite of (2)'s second requirement.
4. The amount of query results in which there were 2 evidence parameters, and non-finite a relative entropy was yielded for ont of the nets given the same query.
5. The sum of the NB nets relative entropy given i evidence parameters.

4.3 Task 4.4

N	M1 wins%	M2 wins%	Sum div M1	Sum div M2	Number of 'inf'
1	1.0	0	9.40655510606201e-15	1.7252847272649334	72
2	0	0	0	0	0
3	0	0	0	0	0
4	0	0	0	0	0

We see that M1 wins for all cases, in which we have the evidence *age* every time and only of length 1.

N	M1 wins%	M2 wins%	Sum div M1	Sum div M2	Number of 'inf'
1	1.0	0	3.126e-15	0.815	52
2	0.833	0.167	0.179	0.817	46
3	0.5	0.5	0.582	0.519	53
4	0.0	1.0	0.103	0.0	49

With an increasing number of priors, we see that the second model becomes better, and when all priors are conditions, it wins 100 % of the times for those cases in which the the set of conditioned priors exists in the data. The behavior is not unexpected as for the second model, that being the NB net with reversed edges, we have all priors knows in the final case. Thus, the model can in this case be regarded as probabilistic regression where we remove drop combinations for which we do not have data.

4.4 Task 4.5

It does not, which isn't a surprise, as in the second model, when our target was *delay*, all parent nodes were conditionally independent. What surprises me is that the NB net is not stronger in this case and it leads me to think that something is erroneous. Given that *delay* is now evidence, we assume the NB net features (i.e the children of *delay*) to be conditionally independent, thus regardless of the number of evidence parameters, it should win, as we would only look at $P(age|delay)$.

5 Task 5

5.1 Task 5.1

Firstly, we take the variables and evidence from our pickled data and look if the variable exist in our training data, then we calculate the relative frequency in the validation data, given the pickled data evidence. Further we then approximate the probability distribution with our model for the variable and evidence in the pickled data, finally calculating the KL-divergence for the models approximated result on the training data, compared to the relative frequency in the validation data. The relative frequency should then be compared to the relative frequency of the pickled data, if those are equal, we have probably overfitted our model.

5.2 Task 5.2

N	M1 wins 100% train	m1 wins 75%	M2 wins 100%	M2 wins 75%
1	0.9213	0.6649	0.0787	0.3351
2	0.7451	0.6875	0.2549	0.3125
3	0.6000	0.6190	0.4000	0.3810
4	0.6000	0.5714	0.4000	0.4286

5.3 5.3

Comparing the table in section *5.2* to the one in section *4.4* we see that the latter tells us that M1 wins 100% of the times given 1 evidence parameter, while M2 wins 100% when having all 4. These results are not reflected in the above table and thus we can draw the conclusion that we do indeed overfit in the previous tasks as we do not split the data into training and validation sets. Also in the above table, it seems that the NB net is more prone to overfitting than the net with reversed edges.

6 Task 6

6.1 Task 6.1

The first model is best according to the scores as it is the one yielding the highest score, i.e $ArgMax(scores) = Model1$.

6.2 Task 6.2

The main idea is to instead of knowing the graph structure, we can evaluate multiple structures given the data. From a Baysian point of view this will be to compute the posterior over graph structures ($P(m|\mathbf{D})$) where m is a certain model. The score is then decided by the model evidence $P(\mathbf{D}|m)$. [1, p.418] By learning the structure instead of assuming the structure, we can also learn dependencies within the graph that shows underlying relations between variables. With the K2-divergence, we look at estimating a statistical model for the underlying dependencies. Doing this, we obtain a network structure that is more general than if we use our assumed structure. This way, the structure is less likely to overfit and thus should yield better results for new data. [3, p.784,785,799]

6.3 Task 6.3

The best model, given the search result

$$('avg\_cs','avg\_mat'),('avg\_cs','age'),('avg\_mat','delay')$$

using the chain rule, we obtain:

$$\begin{aligned} P(avg\_cs, avg\_mat, age, delay) = \\ P(avg\_cs)P(avg\_mat|avg\_cs)P(age|avg\_cs)P(delay|avg\_mat) \end{aligned} \tag{8}$$

and the conditional in-dependencies are

$$\begin{aligned} age \perp avg\_mat|avg\_cs \\ avg\_cs \perp delay|avg\_mat \end{aligned} \tag{9}$$

6.4 Task 6.4

Using Hill Climb Search on the data for 4 variables, we obtain the same graph structure, results for 5 variables are the same. Guessing that implies that gender is marginalized out? Using **BicScore** instead of **K2Score** we do not obtain the same graph. Instead, the graph is factorised as:

$$('avg\_cs','avg\_mat'),('avg\_mat','delay')$$

From the definition of **BicScore** we read that "*Computes a score that measures how much a given variable is "influenced" by a given list of potential*

parents." As we know that we lack data for all possible combinations, it seems that **BicScore** results in a simplified model that is only influenced by that data points at hand, which is likely to lead to worse generalization.

6.5 Task 6.6

I feel that using PGMs gives a simple understanding of the data structure, conditional independencies and inference. Though these models have not been very complex, I see great value in using PGMs for more complex problems.

References

[1] C. M. Bishop. *Pattern recognition and machine learning.* Springer, 2009.

[2] T. Prof. Fomby. NaÏve bayes classifier. 2008.

[3] D. Friedman N. Koller. *Probabilistic Graphical Models principles and techniques.* The MIT press, 2009.