ELEN-0060: Information and Coding Theory

Project 1 - Information Measures

Maxime Goffart
180521

Olivier Joris
182113

Academic year 2021 - 2022

1 Implementation

1.1 Function `entropy`

We are using this mathematical formula:

$$\mathcal{H}(\mathcal{X}) = -\sum_{X_i} P(X_i) \log_2 P(X_i)$$

Our implementation is first filtering the negative $P(X_i)$ because the $\log(x)$ function is only defined for $x \in]0, +\infty[$. Then, it is performing the sum over all X_i using the `sum` method of the numpy library and return the result according to the above mathematical formula.

Intuitively, the entropy is measuring the expected amount of information provided when observing an event corresponding to one value of the random variable. The less probable an event is, the more the amount of information provided is when this event occurs.

1.2 Function `joint_entropy`

We are using this mathematical formula:

$$\mathcal{H}(\mathcal{X}, \mathcal{Y}) = -\sum_{X_i, Y_j} P(X_i \cap Y_j) \log_2 P(X_i \cap Y_j)$$

We notice that this joint entropy is equal to the previously defined entropy taking as argument the joint distribution of the two random variables instead of the marginal distribution of the single random variable.

It is why we are implementing this function by simply reshaping the joint probability distribution to a one dimensional array. In this way, we can just call the `entropy` function with this reshaped array as argument.

1.3 Function `conditional_entropy`

We are using this mathematical formula which has been demonstrated in the theoretical course:

$$\mathcal{H}(\mathcal{X}|\mathcal{Y}) = \mathcal{H}(\mathcal{X}, \mathcal{Y}) - \mathcal{H}(\mathcal{Y})$$

Our implementation consists thus in computing the marginal probability distribution $P_{\mathcal{Y}}$ by marginalizing the joint probability distribution P_{xy} over $\mathcal{X}$. This is done using the `sum` function of the numpy library using `axis = 0` as argument. Then, we just compute the result using the above mathematical formula and the previous defined functions.

An equivalent way to compute this property is to directly use the mathematical formula of the entropy:

$$\mathcal{H}(\mathcal{X}, \mathcal{Y}) = -\sum_{X_i, Y_j} P(X_i \cap Y_j) \log_2 P(X_i|Y_j)$$

1.4 Function `mutual_information`

We are using this mathematical formula which has been demonstrated in the theoretical course:

$$\mathcal{I}(\mathcal{X};\mathcal{Y}) = \mathcal{H}(\mathcal{X}) - \mathcal{H}(\mathcal{X}|\mathcal{Y})$$

Our implementation consists thus in computing the marginal probability distribution $P_{\mathcal{X}}$ by marginalizing the joint probability distribution P_{xy} over $\mathcal{Y}$. This is done using the `sum` function of the numpy library using `axis = 1` as argument. Then, we just compute the result using the above mathematical formula and the previous defined functions.

The mutual information between two discrete random variables measures the level of dependence between these random variables. Especially, if $\mathcal{I}(\mathcal{X};\mathcal{Y}) = 0$, $\mathcal{X}$ and $\mathcal{Y}$ are independent.

1.5 Functions `cond_joint_entropy` and `cond_mutual_information`

We are using these mathematical formulas which have been demonstrated in the theoretical course:

$$\mathcal{H}(\mathcal{X},\mathcal{Y}|\mathcal{Z}) = \mathcal{H}(\mathcal{X},\mathcal{Y},\mathcal{Z}) - \mathcal{H}(\mathcal{Z})$$

$$\mathcal{I}(\mathcal{X};\mathcal{Y}|\mathcal{Z}) = \mathcal{H}(\mathcal{X}|\mathcal{Z}) + \mathcal{H}(\mathcal{Y},\mathcal{Z}) - \mathcal{H}(\mathcal{X},\mathcal{Y},\mathcal{Z})$$

Our implementations consist thus in computing the right marginal probability distributions by marginalizing the joint probability distribution P_{xyz}. This is done as in the previous sections using the `axis` argument of the `sum` function of the numpy library. Then, we just compute the result using the above mathematical formulas and the previous defined functions.

2 Weather forecasting

2.1 Entropy and cardinality of each variable

The corresponding entropy and cardinality of each variable can be observed in the table 1.

Random variable $\mathcal{X}$	**Entropy $\mathcal{H}(\mathcal{X})$**	**Cardinality**
temperature	1.511	4
air_pressure	1.000	2
same_day_rain	1.475	3
next_day_rain	1.569	3
relative_humidity	1.000	2
wind_direction	2.000	4
wind_speed	1.585	3
cloud_height	1.585	3
cloud_density	1.584	3
month	3.583	12
day	2.806	7
daylight	0.999	2
lightning	0.325	3
air_quality	0.536	3

Table 1: Entropy $\mathcal{H}(\mathcal{X})$ and cardinality of each random variable $\mathcal{X}$.

We see in this table that the higher the cardinality, the higher the entropy. This can be theoretically justified by the fact that the entropy of a instrument corresponds to the amount of information gained when its value is known. It is why with a larger cardinality the entropy is larger. Indeed, the instrument can take more values than with lower cardinalities which implies that the probability of getting a specific value for a variable is smaller thus the entropy is higher.

2.2 Conditional entropy of `next_day_rain` given each of the other variables

The corresponding conditional entropy of each variable with `next_day_rain` can be observed in the table 2.

Random variable $\mathcal{Y}$	**Conditional entropy $\mathcal{H}(\texttt{next\_day\_rain}\|\mathcal{Y})$**
temperature	1.568
air_pressure	0.940
same_day_rain	1.389
relative_humidity	1.301
wind_direction	1.568
wind_speed	1.568
cloud_height	1.567
cloud_density	1.567
month	1.565
day	1.567
daylight	1.568
lightning	1.568
air_quality	1.568

Table 2: Conditional entropy $\mathcal{H}(\texttt{next\_day\_rain}|\mathcal{Y})$ with each random variable $\mathcal{Y}$.

(a) When the conditioning variable is `wind_direction`, the conditional entropy is nearly equal to the entropy of the `next_day_rain` variable without any conditioning. It can be explained by the fact that knowing the `wind_direction` variable does not really provide any additionnal information on the `next_day_rain` variable which seems intuitively logical.

(b) When the conditioning variable is `same_day_rain`, the conditional entropy is lower than the entropy of the `next_day_rain` variable without any conditioning. It can be explained by the fact that knowing the `same_day_rain` variable provides additionnal information on the `next_day_rain` which seems intuitively logical. The provided amount of information corresponds here to 1.568 - 1.389 = 0.179 bits.

2.3 Mutual information between two variables

We can deduce that the variables `relative_humidity` and `wind_speed` are independent because their mutual information is equal to 0.

We can deduce that the variables `month` and `temperature` are dependent because their mutual information is > 0 (here it is equal to 0.535).

2.4 Choose a single instrument

Based on the mutual information, the variable kept would be air_pressure because the amount of mutual information between this variable and next_day_rain is the highest compared to all the other variables (it is equal to 0.629).

Based on the conditional entropy, the variable kept would be air_pressure because the entropy of next_day_rain with this variable is the lowest compared to all the other variables (it is equal to 0.940). The variable kept does not change from the one chosen using the mutual information.

2.5 Deletion of the dry sample of the next_day_rain variable from the dataset

Based on the mutual information, the variable kept would be relative_humidity because the amount of mutual information between this variable and next_day_rain is the highest compared to all the other variables (it is equal to 0.439).

Based on the conditional entropy, the variable kept would be relative_humidity because the entropy of next_day_rain with this variable is the lowest compared to all the other variables (it is equal to 0.560). The variable kept does not change from the one chosen using the mutual information.

If we delete the dry sample of the next_day_rain variable, the variable kept becomes the relative_humidity.

2.6 Instrument kept if we have a thermometer for free

Based on the conditional mutual information, the variable kept would be air_pressure because the conditional mutual information between this variable, next_day_rain and this variable knowing temperature is the highest compared to all the other variables (it is equal to 0.629). This does not change the kept variable according to the question 9.

Based on the conditional joint entropy, the variable kept would be lightning because the conditional joint entropy of next_day_rain and this variable knowing temperature is the lowest compared to all the other variables (it is equal to 1.821). This changes the kept variable according to the question 9.