INF-2600: Artificial Intelligence, AI - Methods and applications

Assignment 3: Analyzing Sensor Data for Weather Prediction

Amund Harneshaug Strøm

UiT id: ast283@uit.no

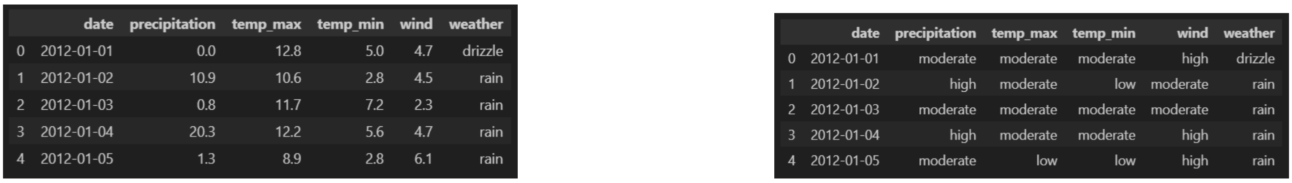
May 9, 2024

1 Task 1: Bayesian Network for Weather Prediction.

1.1 Task 1.1:

To construct a Bayesian Network that models the relationships between different weather variables required mainly the use of the *pandas* and *pgmpy* python library.

The *pandas* library is used to import and handle the weather data, it creates a dataframe which makes it easier to handle the data in the weather dataset. The dataframe consist of these variables; date, precipitation, max temperature, minimum temperature, wind, and weather. This dataframe is used for two major reason; to create bounds for the continuous labels, and calculate the probabilities in the dataset. The bounds is used to separate the continuous variables into different categories according to their value compared to the rest of the dataset. This is important since it makes it possible to create conditional probability distributions later on. These continuous variables are precipitation, max temperature, minimum temperature, and wind, and their value labels range from low, moderate, and high.



	date	precipitation	temp_max	temp_min	wind	weather
0	2012-01-01	0.0	12.8	5.0	4.7	drizzle
1	2012-01-02	10.9	10.6	2.8	4.5	rain
2	2012-01-03	0.8	11.7	7.2	2.3	rain
3	2012-01-04	20.3	12.2	5.6	4.7	rain
4	2012-01-05	1.3	8.9	2.8	6.1	rain

(a) Dataframe with values.

	date	precipitation	temp_max	temp_min	wind	weather
0	2012-01-01	moderate	moderate	moderate	high	drizzle
1	2012-01-02	high	moderate	low	moderate	rain
2	2012-01-03	moderate	moderate	moderate	moderate	rain
3	2012-01-04	high	moderate	moderate	high	rain
4	2012-01-05	moderate	low	low	high	rain

(b) Dataframe with lables.

Figure 1: The dataframe before and after being converted to labels.

The last major reason to use the *pandas* library is to calculate the probabilities for each variable in the dataset. To figure out how to calculate the probabilities correctly, we must look at the hierarchy of the Bayesian Network.

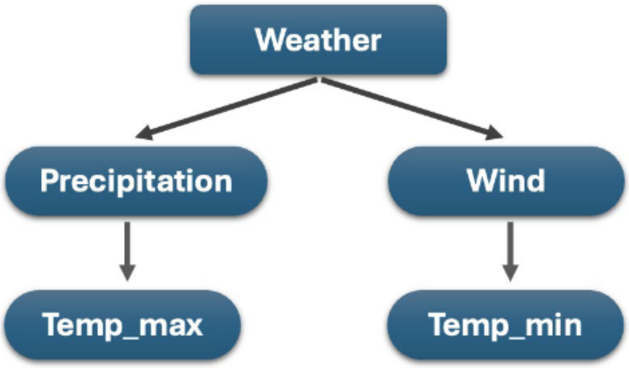


Figure 2: Hierarchy of the Bayesian Network. Source: [2]

As seen in Figure 2, weather does not have any parents, so we only need to calculate the marginal probabilities which refer to the probabilities of observing each weather type independently, without considering any other factors or conditions. The remaining variables have different connections so we must calculate the conditional probability accordingly. Conditional probability is the likelihood of one event happening under the condition that another event has already taken place.

To create the Bayesian Network we used the *pgmpy* python library which has a class that creates a Bayesian Network with the given input. As seen Figure 2 we see the hierarchy of the network, and the input to the class consisted of the edges between the nodes. Now that the network has been created, it needs probabilities for each variable/node in the network. These probabilities have been calculated with the help of the *pandas* python library. Before inserting the probabilities into the network we need to define the conditional probability distribution tables (CPD) using the probabilities calculated beforehand. The CPD's can be defined using another class in the *pgmpy* library, using the calculated probabilities as inputs. Now that the CPD's are created, we can simply add them to the Bayesian Network to complete the network.

1.2 Task 1.2

Implement exact inference to answer the following questions. Use Variable Elimination for this.

1.2.1 Task 1.2.1

(a) The probability of high wind when the weather is sunny 0.254%.

(b) Probability of sunny weather when the wind is high: 0.241%.

1.2.2 Task 1.2.2

(a) The most probable condition is:

- Weather : Drizzle
- Wind : Mid
- Precipitation : Mid
- Max Temperature : Mid
- Minimum Temperature : Mid

The result shows the combination that is the most likely to happen in the dataset. Meaning, this exact combination of labels for the variables occurred the most in the dataset.

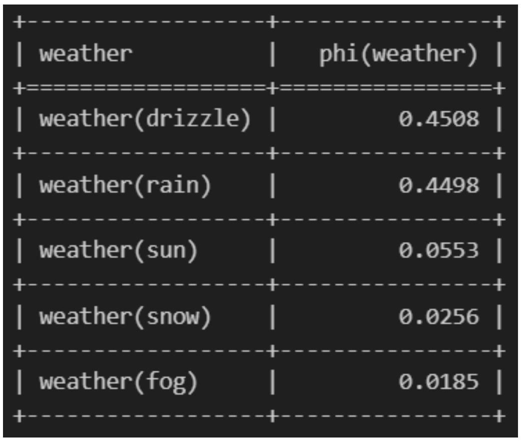
(b) The most probable condition for weather, wind, and precipitation combined is:

- Weather : Drizzle
- Wind : Mid
- Precipitation : Mid

The results show again the most occurred combination of labels, but this time only for the variables weather, wind, and precipitation.

1.2.3 Task 1.2.3

The probability associated with each weather state, given that the precipitation is medium.



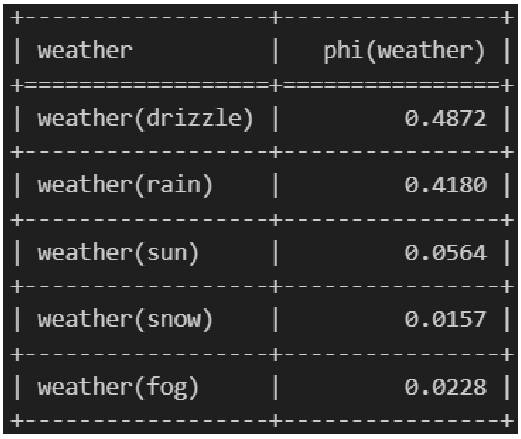
```
+------------------+----------------+
| weather          |   phi(weather) |
+==================+================+
| weather(drizzle) |         0.4508 |
+------------------+----------------+
| weather(rain)    |         0.4498 |
+------------------+----------------+
| weather(sun)     |         0.0553 |
+------------------+----------------+
| weather(snow)    |         0.0256 |
+------------------+----------------+
| weather(fog)     |         0.0185 |
+------------------+----------------+
```

Figure 3: The probability associated with each weather state.

The results is an table that shows the likelihood for each weather state to happen given that precipitation is medium. We see that 'drizzle' is the most likely, which correlates with the result in Task 1.2.2. We also see that the sum of all the probabilities is equal to 1, indicating that one of the weather states must happen.

1.2.4 Task 1.2.4

The probability associated with each weather state, given that precipitation is medium and wind is low or medium.



```
+------------------+----------------+
| weather          |   phi(weather) |
+==================+================+
| weather(drizzle) |         0.4872 |
+------------------+----------------+
| weather(rain)    |         0.4180 |
+------------------+----------------+
| weather(sun)     |         0.0564 |
+------------------+----------------+
| weather(snow)    |         0.0157 |
+------------------+----------------+
| weather(fog)     |         0.0228 |
+------------------+----------------+
```

Figure 4: The probability associated with each weather state.

The method used in this task was to use the *query* method in the *VariableElimination*[1] class, which does a query on the Bayesian Network over the given variables and evidence. The variable that is searched is weather, and the evidence is the given states that must occur i.e. precipitation is medium and wind is low or medium.

With the addition of the wind factor makes the 'drizzle' state even more likely. Which again correlates with what was shown in task 1.2.2. The reason the 'drizzle' state increased is because it is the most likely state whenever precipitation is medium and wind is medium.

1.3 Task 1.3

Use approximate inference methods to answer all four questions in Task 1.2.

1.3.1 Likelihood Weighted Sample: Task 1.2.1

What is the probability of high wind when the weather is sunny?

Exact inference	Likelihood Weighted Sample
0.254%	0.255%

Table 1: The probability of high wind when the weather is sunny.

The table shows that there is almost no difference in Exact inference and Likelihood Weighted Sample. The small change in the results is because approximate inference introduce some level of error or approximation.

What is the probability of sunny weather when the wind is high?

Exact inference	Likelihood Weighted Sample
0.241%	0.069%

Table 2: The probability of sunny weather when the wind is high.

The table shows a significant difference between results, the reason for this is that Likelihood Weighted Sampling might not have sampled the whole dataset to capture the full probability of the event. It also tends to introduce some biases whenever the evidence variables are highly influential on the network's structure, which might be in this case.

1.3.2 Rejection Sampling: Task 1.2.2

Calculate all the possible joint probability and determine the best probable condition.

Variables	Exact inference	Rejection Sampling
Weather	Drizzle	Drizzle
Wind	Mid	Mid
Precipitation	Mid	Mid
Max Temperature	Mid	Mid
Minimum Temperature	Mid	Mid

Table 3: The probability of sunny weather when the wind is high.

What is the most probable condition for precipitation, wind and weather, combined?

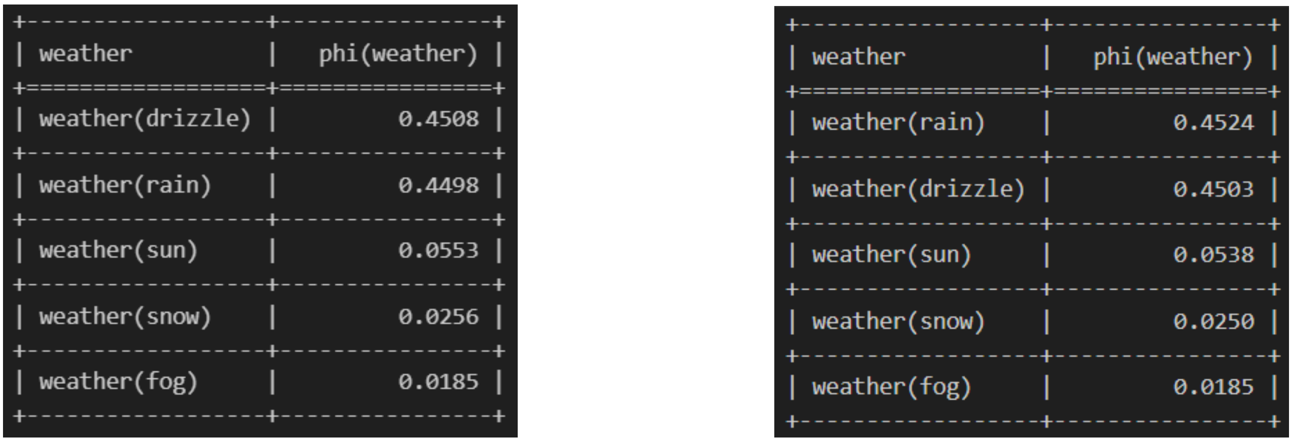
Variables	Exact inference	Rejection Sampling
Weather	Drizzle	Drizzle
Wind	Mid	Mid
Precipitation	Mid	Mid

Table 4: The probability of sunny weather when the wind is high.

Both the results of rejection sampling give the same result as exact inference. This is likely due to the fact that the Bayesian Network is relatively simple, has few variables and a low-dimensional joint probability distribution.

1.3.3 Approx Inference: Task 1.2.3

Find the probability associated with each weather, given that the precipitation is medium?



```
+------------------+----------------+
| weather          |   phi(weather) |
+==================+================+
| weather(drizzle) |         0.4508 |
+------------------+----------------+
| weather(rain)    |         0.4498 |
+------------------+----------------+
| weather(sun)     |         0.0553 |
+------------------+----------------+
| weather(snow)    |         0.0256 |
+------------------+----------------+
| weather(fog)     |         0.0185 |
+------------------+----------------+
```

(a) Exact inference.

```
+------------------+----------------+
| weather          |   phi(weather) |
+==================+================+
| weather(rain)    |         0.4524 |
+------------------+----------------+
| weather(drizzle) |         0.4503 |
+------------------+----------------+
| weather(sun)     |         0.0538 |
+------------------+----------------+
| weather(snow)    |         0.0250 |
+------------------+----------------+
| weather(fog)     |         0.0185 |
+------------------+----------------+
```

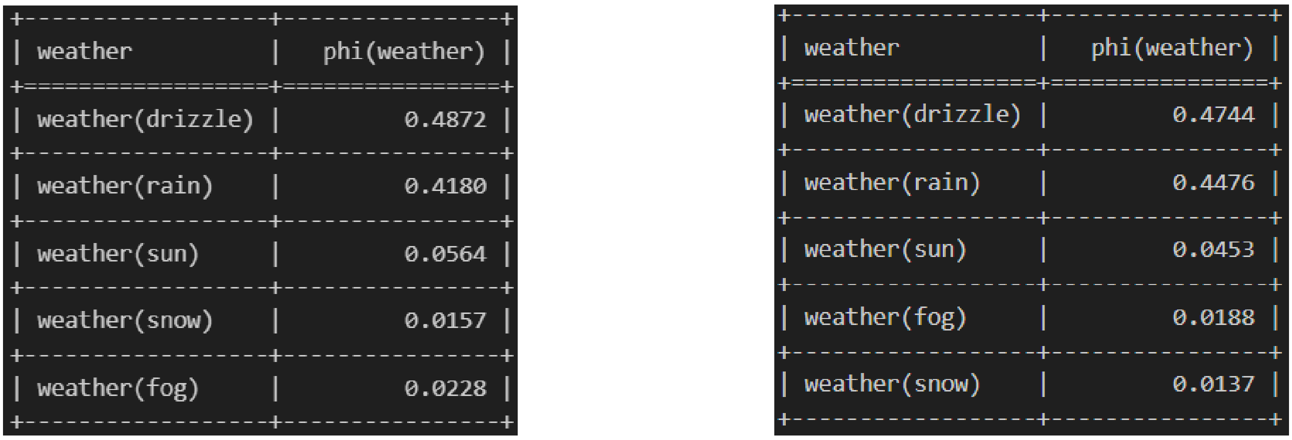
(b) Approx Inference.

Figure 5: The probability associated with each weather, given that the precipitation is medium.

There is a small difference in results since Approximate inference does not always capture the exact probability distribution as effectively as exact inference. Which leads to slight variations in the results as seen in the above figure.

1.3.4 Normal Sampling: Task 1.2.3

What is the probability of each weather condition given that precipitation is medium and wind is low or medium?



```
+------------------+----------------+
| weather          |   phi(weather) |
+==================+================+
| weather(drizzle) |         0.4872 |
+------------------+----------------+
| weather(rain)    |         0.4180 |
+------------------+----------------+
| weather(sun)     |         0.0564 |
+------------------+----------------+
| weather(snow)    |         0.0157 |
+------------------+----------------+
| weather(fog)     |         0.0228 |
+------------------+----------------+
```

(a) Exact inference.

```
+------------------+----------------+
| weather          |   phi(weather) |
+==================+================+
| weather(drizzle) |         0.4744 |
+------------------+----------------+
| weather(rain)    |         0.4476 |
+------------------+----------------+
| weather(sun)     |         0.0453 |
+------------------+----------------+
| weather(fog)     |         0.0188 |
+------------------+----------------+
| weather(snow)    |         0.0137 |
+------------------+----------------+
```

(b) Normal Sampling.

Figure 6: The probability of each weather condition given that precipitation is medium and wind is low or medium.

The above figure shows yet some small differences. The reason for this is that normal sampling provides an estimation that can vary slightly each time it is run, depending on the randomness of the samples generated. While, Exact Inference always returns the same results given the same network and conditions.

1.4 Task 1.4

1.4.1 Task 1.4.1

To create additional Bayesian Networks I had to do some small changes in the code, such as changing the input given in the Bayesian Network class, which determines the structure of the network. Since the structure of the network changed, the calculation of the probabilities also had to change accordingly. As seen in the figure below, the calculation had to account for a node having two different parents. This also resulted in changes in the creation of the tabular conditional probability distribution.

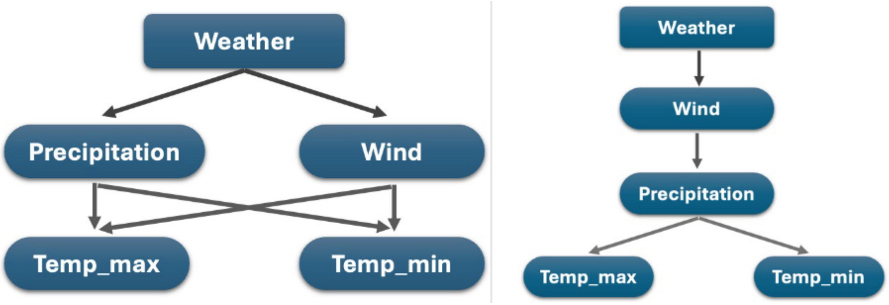


Figure 7: Hierarchy of the additional Bayesian Networks. Source: [2]

1.4.2 Task 1.4.2

The bellow tables show the top 5 most occurred combinations in the different Bayesian Networks, and is calculated using exact inference.

Weather	Precipitation	Wind	Temp max	Temp min	Probability
drizzle	mid	mid	mid	mid	0.1086
rain	mid	mid	mid	mid	0.0931
–:–	–:–	low	mid	mid	0.0538
drizzle	mid	low	mid	mid	0.0495
–:–	–:–	mid	mid	high	0.0477

Table 5: Top 5 combinations in the default hierarchy.

Weather	Precipitation	Wind	Temp max	Temp min	Probability
drizzle	mid	mid	mid	mid	0.1048
rain	mid	mid	mid	mid	0.0899
–:–	–:–	low	mid	mid	0.0577
drizzle	mid	low	mid	mid	0.0490
–:–	–:–	mid	mid	high	0.0486

Table 6: Top 5 combinations in the hierarchy 1.

Weather	Precipitation	Wind	Temp max	Temp min	Probability
drizzle	mid	mid	mid	mid	0.1022
rain	mid	mid	mid	mid	0.0877
–:–	–:–	low	mid	mid	0.0574
drizzle	mid	low	mid	mid	0.0487
–:–	–:–	mid	mid	high	0.0365

Table 7: Top 5 combinations in the hierarchy 2.

As seen in the above tables, all the different hierarchies have the same 5 most occurred combinations, but their probabilities varies slightly. The bellow table only shows the probabilities of combinations in the same order as the above tables.

Default	Hierarchy 1	Hierarchy 2
0.1086	0.1048	0.1022
0.0931	0.0899	0.0877
0.0583	0.0577	0.0574
0.0495	0.0490	0.0487
0.0477	0.0486	0.0365

Table 8: Top 5 combinations and their respective probabilities.

The reason we see these changes to the probabilities is because slight changes to the Bayesian Network will lead to unforeseen changes in the Conditional Probability Tables, which will then cause the probabilities to change. Even though the we see changes in the probabilities we did not see any changes to the combinations. This is because the networks still use the same data and you could therefore expect to see the same top combinations.

2 Task 2: Hierarchical Feature Analysis in Bayesian Networks

2.1 Task 2.1

To employ Bayesian Network structure learning algorithms to construct a Bayes Net model you need to choose an structure learning algorithm that suits your specific needs. The task of structure learning for Bayesian networks refers to learning the structure of the directed acyclic graph (DAG) from data. The term DAG consists of 3 terms;

- Directed, which means the graph includes edges with directions.
- Acyclic, which indicates that the graph has no cycles, meaning you cannot start at one node and follow a sequence of edges that eventually loops back to the starting node.
- Graph, a structure consisting of nodes and edges

There are two major approaches for structure learning: score-based and constraint-based [3].

The base concept of an constraint-based approach is to identify variables that are conditionally independent or dependent, and use this information to construct the network structure. This approach assumes that the underlying structure of the data can be represented by a DAG. It perform a series of statistical tests to check for independence between nodes, followed by set of conditions that renders the pair independent [3]. This process is repeated and until it is satisfied. The advantage of this approach is that it does not require any assumption about the structure of the dataset prior, and is only based on observed data through statistical testing.

The score-based approach evaluate the fit of different networks to the data using a scoring function. It searches over possible DAG structures and picks the one that maximizes the scoring function [3]. Picking the right scoring function is crucial, these are some of the most common; Bayesian Information Criterion, Akaike Information Criterion, and Bayesian Scoring Criterion. An equally important task is to effectively search over the space of all possible DAGs, there are also several strategies for this and the most common is local search- and greedy search-algorithms[3]. The advantages of this approach is that it can include different constraints into the scoring function, and the search algorithms may discover complex fits for the model.

With these different approaches in mind, I would pick an score-based approach since it may find an unique DAG that fits the model that would be otherwise difficult to find by hand. Also since I don't have an in-depth knowledge about the dataset, it would be better for an algorithm to find different connections between nodes. Considering the size of the dataset I think this approach wont be too computational expensive, and therefor it should be possible to find an DAG structure for the dataset.

2.2 Task 2.2

Once you have established an connection between the nodes in the Bayesian Network, the next step is to learn the CPDs that capture the probabilistic dependencies between the nodes. There are several approaches to this, that all depends on the data you are trying to capture. By taking a quick look at the given data in Norwegian maritime sector (SPRICE), it looks like the data is mostly complete and there is no missing fields. Therefore Maximum Likelihood Estimation is probably sufficient. This method has become dominant within the fields of statistical inference, and preforms well whenever the given data is complete [4].

2.3 Task 2.3

The figure below is an visualization of the constructed Bayesian Network used in this task.

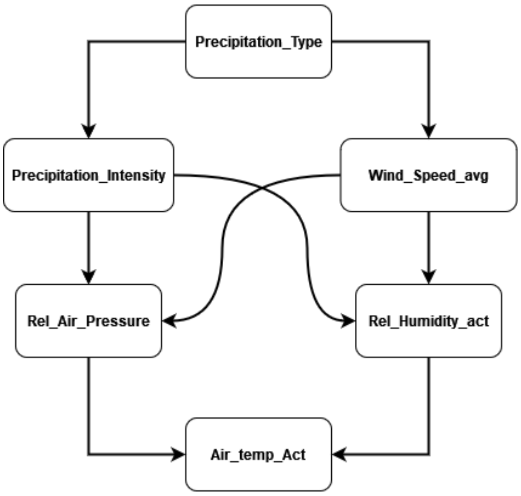


Figure 8: Visualization of the Bayesian Network created in task 2.3.

The variables in this network is chosen from the highlighted features given in the assignment text. It does not include the variable “Wind_Direction_vct”, as I did not see it as useful for the network. The structure of the network is an extension of the network given in Task 1.4.1, and each variable is placed in the network according to my intuition and own reasoning. For example it made sense for me to place “Air_temp_Act” at the bottom, since the temperature is mainly influenced by other variables.

There is a python file in the “src” folder that computes similar inferences as Task 1.1, 1.2, and 1.3 on the constructed Bayesian Network. There is however an issue with the Approximate Inference tasks, as it seems that the probability does not sum up to 1. I have tried my best to debug this, but to no avail. According to the an check in the code the model should be consistent, which makes it much harder to understand the problem. Regardless, the Variable Elimination computations work without any issues.

References

[1] Ankur Ankan. Variable Elimination. In Pgmpy, May 06, 2024, from `https://pgmpy.org/exact_infer/ve.html`

[2] Assignment 3: Analyzing Sensor Data for Weather Prediction, May 06, 2024.

[3] Volodymyr Kuleshov & Stefano Ermon. Structure learning for Bayesian networks. May 06, 2024, from `https://ermongroup.github.io/cs228-notes/learning/structure/`

[4] Wikipedia contributors. Maximum likelihood estimations. From Wikipedia, the free encyclopedia. May 06, 2024, from `https://en.wikipedia.org/wiki/Maximum_likelihood_estimation`