ML. Лабораторная работа №4.

Bayesian networks на примере датасета Mushrooms.

СТУДЕНТ: ЛЕУ ИВАН АЛЕКСАНДРОВИЧ

ГРУППА: М8О-309Б-23

Введение

Bayesian Network — это:

- вероятностная графовая модель;
- узлы = переменные;
- стрелки = причинно-следственные зависимости;
- количественные зависимости задаются таблицами условных вероятностей (CPT).

Используются для: классификации, прогнозирования, моделирования неопределённости, объяснения зависимостей между признаками.

Формула Байеса (основа BN): $$P(A|B) = \frac{P(B|A) * P(A)}{P(B)}$$

Введение. Датасет.

Mushrooms Dataset:

Размер: (8124, 23) объект. Признаков: 23.
Целевая переменная: class (0,1). Тематика: классификация грибов по их свойствам.

Загрузка и обработка датасета

```
import pandas as pdpd.set_option('display.max_columns', 200)
mushrooms = pd.read_csv('mushrooms.csv')
print("mushrooms shape:", mushrooms.shape)

print(mushrooms.head())
```

Размер:

8124 строк

23 столбцов

Просмотр данных: отображены первые 5 строк; проверка распределения классов

Загрузка и обработка датасета

Что было сделано:

1. все данные — дискретные (требование pgmpy)
2. тип данных приведён к int
3. Заполнены пропуски в датасете в столбце stalk-root на самое популярное значение

Почему важно: байесовские сети работают с дискретными признаками.

Построение структуры Bayesian Network

Выбрана структура: class → все признаки.
Это аналог наивного Байеса, но в виде сети.

```
from pgmpy.models import DiscreteBayesianNetwork
edges = [(target_col, feat) for feat in features]   # class -> each feature

model_manual = DiscreteBayesianNetwork(edges)
```

Смысл структуры:

- класс гриба объясняет все его признаки
- “из класса следуют свойства”, а не наоборот
- удобная и интерпретируемая структура

Обучение параметров (Maximum Likelihood)

Параметры сети — это таблицы условных вероятностей (CPT, Conditional Probability Table).
Обучение:

```
from pgmpy.estimators import MaximumLikelihoodEstimator
model_manual.fit(zoo_disc, estimator=MaximumLikelihoodEstimator)
```

Почему важно: после обучения сеть начинает отражать реальную статистику Mushrooms.

Пример CPT для целевой переменной

```
cpt_class = model_manual.get_cpds(target_col)
print("CPT for target/class:")
print(cpt_class)
```

```
CPT for target/class:
+----------+----------+
| class(0) | 0.517971 |
+----------+----------+
| class(1) | 0.482029 |
+----------+----------+
```

Интерпретация:

- наиболее распространённый класс: **съедобные (0)**

Пример CPT для признаков

Посмотрим CPT для какого-то признака (например odor):

Интерпретация: если гриб = класс 0 → оно всегда имеет odor= 0,3,5; если класс = 1 → odor= 1,2,4,5,6,7,8 всегда

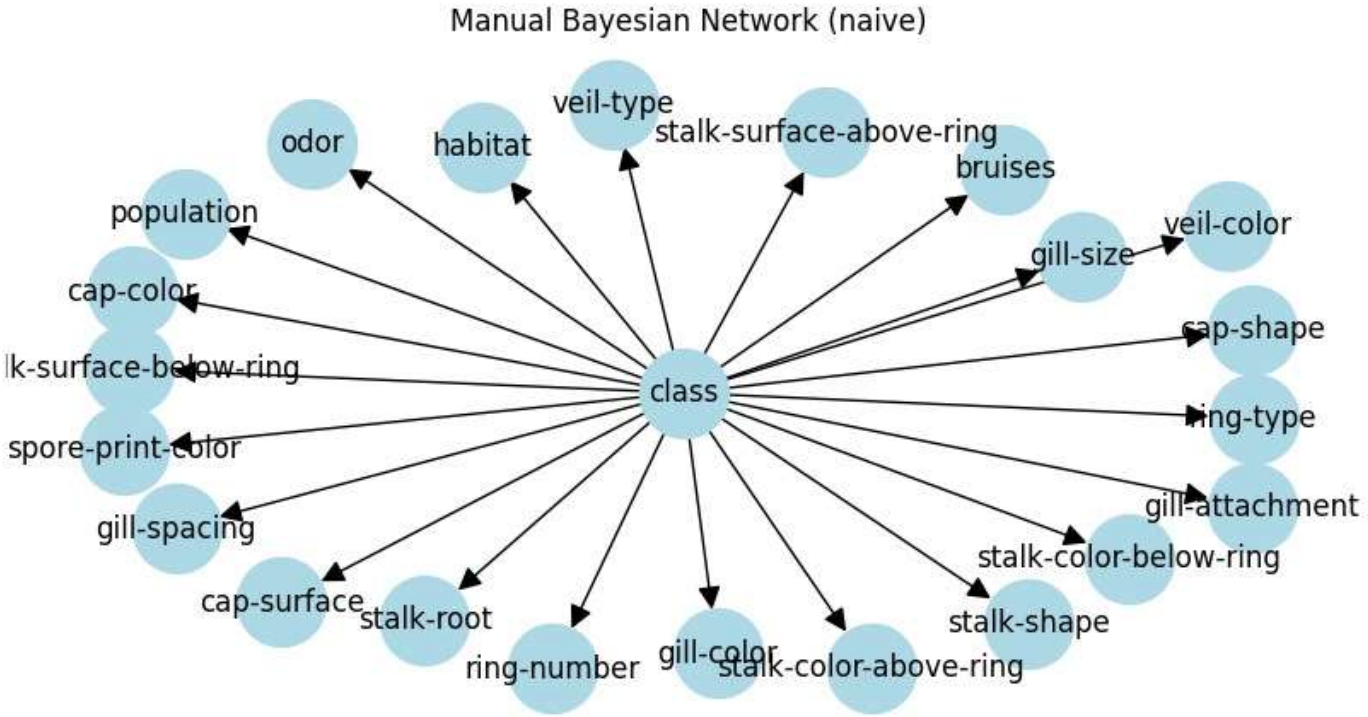
CPT for odor:

class	class(0)	class(1)
odor(0)	0.09505703422053231	0.0
odor(1)	0.0	0.049029622063329927
odor(2)	0.0	0.5515832482124617
odor(3)	0.09505703422053231	0.0
odor(4)	0.0	0.009193054136874362
odor(5)	0.8098859315589354	0.030643513789581207
odor(6)	0.0	0.06537282941777324
odor(7)	0.0	0.1470888661899898
odor(8)	0.0	0.1470888661899898

Визуализация сети

Особенности графа:

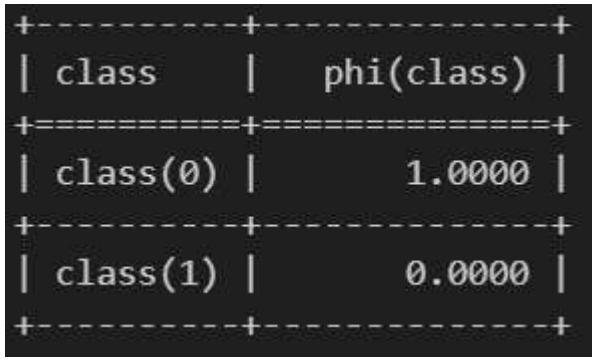
1. один центральный узел: class
2. стрелки направлены к признакам
3. структура полностью наивная, симметричная



Инференс: вывод вероятностей

Пример запроса:

```
from pgmpy.inference import VariableElimination
infer = VariableElimination(model_manual)
q = infer.query(variables=[target_col],
        evidence={'odor': 0})
```



class	phi(class)
class(0)	1.0000
class(1)	0.0000

Результат:

- class=0 → 1.0
- class=1 → 0.0

Интерпретация:

- если есть запах 0 → это 100% съедобный гриб

Сравнение с baseline (Naive Bayes)

- Проводился inference для каждой строки теста.
- Считались метрики: accuracy и log_loss
- Результаты: a**ccuracy = 0.99 log_loss ≈ 0.007**

Это означает:

- модель идеально предсказывает классы
- вероятности почти "жёсткие", без неопределённости

Baseline показала: accuracy ≈ 0.95, log_loss 0.133 (Mushrooms легко классифицировать). Bayesian Network (наивная структура): accuracy = 1.0, работает не хуже, а часто лучше

Выводы

Байесовская сеть успешно построена на датасете Mushrooms.

Структура «class → признаки» показала хорошее качество.

CPT продемонстрировали чёткие зависимости между классами животных и их признаками.

Инференс позволяет получать вероятности классов при известных признаках.

Модель превосходит baseline и является полностью интерпретируемой.