Introduction

What PGMs are, what this course is about and how it is organised.

Wilker Aziz

w.aziz@uva.nl

https://probabll.github.io

Outline and goals

This class is a conceptual introduction to probabilistic graphical models (PGMs), it also covers course organisation.

ILOs After this class the student

- understands what PGMs are about at a conceptual level;
- recognises what PGMs are meant for;
- knows what this course will cover;
- knows how this course is organised.

Textbook for this course: Koller and Friedman [1].

Table of contents

Probabilistic Graphical Models

Probabilistic Graphical Models (PGMs)

Probabilistic ***graphical models*** combine

- *statistics and probability theory*
- and ***computer science***

to solve hard problems involving variables we are **uncertain** about.

Early applications

Health (e.g., diagnosis)

Computer vision (e.g., image segmentation)

Natural language processing (e.g., named-entity recognition)

Early applications

Health (e.g., diagnosis)

Computer vision (e.g., image segmentation)

Natural language processing (e.g., named-entity recognition)

What do these problems have in common?

- They have a very large number of variables that we have to reason about.
- There is going to be significant uncertainty about the right answer, no matter how clever an algorithm we design.

The **model** is a declarative representation of our understanding of the world.

It is expressed in terms of the variables we are reasoning about and how we believe or assume they interact with one another.

PGMs are designed to help us deal with uncertainty.

The model itself is a *representation* of our uncertainty about the world. And it can be used to *support decision making*.

Uncertainty comes in many forms and for many reasons:

- Partial knowledge of the state of the world
- Noisy observations
- Modelling limitations
- Inherent ambiguity and/or stochasticity

Probability theory is the most widely used framework for uncertainty representation.

For other mathematical frameworks for uncertainty repreentation see **?**].

Graphs allow us to represent complex systems involving many interacting variables.

We will use graphs to code domain knowledge and represent probability distributions.

Graphs are not only efficient data structures that support efficient algorithms, they are amenable to systematic analysis: we can ascertain properties of distributions by studying the graph structure.

Summary

What is a probabilistic graphical model?

A representation of the world that takes into account the things we care and are uncertain about and our assumptions about their relationships.

What can we do with it?

Answer queries about the world and make decisions under uncertainty.

How do we construct them?

A combination of domain knowledge and learning from historical data.

Two Examples

TODO: BN example (probably medical dianosis)

TODO: MN example (probably image segmentation)

What are you going to learn?

Representation

- Directed and undirected graphical models
- Parameterisation

Reasoning

- Exact and approximate inference
- Decision making

Learning

- Parameters
- Observed and unobserved data

Course Organisation

Learning Activities

HCs: we develop the main theory planned for the week and solve some exercises; sometimes there's reading that's necessary before or after the class.

Learning Activities

HCs: we develop the main theory planned for the week and solve some exercises; sometimes there's reading that's necessary before or after the class.

LCs: we translate the main ideas in PGMs to Python code; because PGMs deal with graph structure, probabilistic inference and statistical learning, programming exercises will often help you consolidate your knowledge of the theory.

Learning Activities

HCs: we develop the main theory planned for the week and solve some exercises; sometimes there's reading that's necessary before or after the class.

LCs: we translate the main ideas in PGMs to Python code; because PGMs deal with graph structure, probabilistic inference and statistical learning, programming exercises will often help you consolidate your knowledge of the theory.

WCs: we work on exam-like exercises, discuss feedback, and sometimes cover some side topics that help you get a good hold of the material.

Assessment

Exams: midterm (40%) and endterm (40%).

Programming: weekly assignments (due on Friday), the best 5 (out of 6) contribute to 20% of your final grade.

In-class bonus: weekly exercises offered in-class (HC) only, each week we offer 0.5 points for a total of 3 points throughout the entire course; these points go directly to your final grade.

Exam-like questions: these are available for self-study and self-assessment; a subset of these are available as 0-points assignments on ANS, if you submit you get individual feedback.

Grade

Your final grade is obtained as follows:

$$\text{grade} = \mathsf{SIS}(0.8 \times \text{exam} + 0.2 \times \text{programming} + \text{bonus})$$

This grade is rounded and clipped for SIS as follows

- lowerbounded by 1 (e.g., 0.2 is mapped to 1);
- rounded to the closest half point if between 1 and 5 or between 6 and 10 (e.g., 4.2 is mapped to 4.0, 6.8 is mapped to 7);
- rounded to the closest point if it falls between 5 and 6 (e.g., 5.4 is mapped to a 5, while 5.5 is mapped to a 6).

No minimum requirement on any one of the partial grades, but the final grade must be a 6.0 for you to pass this course.

Disclaimer: check *https://canvas.uva.nl/courses/53146/pages/assessment-and-grade* for the official rules, this is only a summary.

Platforms

The course is managed entirely on Canvas, this is the official source of information, announcements, deadlines, and material: *https://canvas.uva.nl/courses/53146*

We use a few additional tools, for specific purposes:

- ANS: exam-like exercises, grading and feedback
- Ed: asynchronous Q&A (see Ed House Rules on Canvas) *https://canvas.uva.nl/courses/53146/pages/ed-house-rules?module_item_id=2640464*

Class recordings are available from Canvas, see Webcolleges: *https://canvas.uva.nl/courses/53146/external_tools/21068*

What Next?

Rest of HC1a: BNs (semantics)

LC1: BNs in code

HC1b: BNs (reasoning and influence)

WC1: BN exercises (semantics, reasoning and influence)

References

[1] Daphne Koller and Nir Friedman. *Probabilistic graphical models: principles and techniques*. MIT press, 2009.