

# PGMs week 1

# Factor Operations

This week we focus on getting our infrastructure going, and exploring the manipulation of discrete probability factors (on their own, i.e. without combining them in some graphical structure).

## 1 Downloads and infrastructure – see the info sheet for details.

- Get the Barber book or pdf.
- Locate Koller's PGM lectures.
- Install either `pgmpy` or `emdw` on you computer.

## 2 Introduction and review

- Watch the *PGM2 Motivation (19:18)* Koller video to get an idea of what probabilistic graphical models are about. We will mostly, but not exclusively, cover the same material as that discussed in this video.
- Review basic probability theory: Consult a standard probability theory textbook (such as PZ Peebles) and make sure that you are comfortable with the following:
  - **Random variables** – both discrete and continuous (RVs)
  - **Probability distributions** – both probability mass functions (PMF) and probability density functions (PDF)
  - **Joint distributions** and how it describes the behaviour of multiple random variables
  - **Conditional distributions** and how you obtain them from joint distributions (chain or product rule)
  - **Marginal distributions** and how you obtain them from joint distributions (sum rule)
  - **Bayes rule** – which will get a life of its own in this work
  - **Statistical independence** – both the definition and its interpretation
- Read Barber chapter 1: Probability refresher, Probabilistic reasoning as a generalisation of logic, prior, likelihood and posterior.

## 3 **WATCH the *PGM3 Preliminaries Distributions(04:57)* and *PGM4 Preliminaries Factors(06:40)* Koller videos on factor operations:**

- Note how a factor product is determined and how this can be used to build a joint distribution.
- Note how conditional and marginal distributions are implemented via factors.
- These examples are for discrete variables. Muse a bit about what might be involved in defining their continuous counterparts.

## 4 DO a simple paper exercise

This one is simple enough that the answers are nearly self-evident, you can easily check against your common sense (please do this!). The objective here is to confirm that you can systematically get to those self-evident answers by using the factor operations you saw in the videos (products, marginalisations and conditions/evidence).

Take two balanced coins. For both of them, mark one side with the number '0' and the other with the number '1'. Now consider casting them and denote their outcomes by letting the random variables  $X$  and  $Y$  take on the values on the facing sides of the coins. Let  $Z = X + Y$ . Make use of the factor operations you encountered earlier on and do the following on paper:

- Define the three factors  $p(X)$ ,  $p(Y)$  and  $p(Z|X, Y)$ .
- Determine the joint  $p(X, Y, Z)$ .
- Determine the marginal  $p(X, Z)$ .
- Determine the conditional  $p(Z|X = 1)$ .
- Determine the conditional  $p(X|Z = 2)$ .
- Determine the marginal  $p(Z)$ .
- Convince yourself that the above really only is a nicely crystallised version of the classical operations you reviewed earlier on.

## 5 DO a simple coding exercise

We will use the `pgmpy DiscreteFactor` or `emdw DiscreteTable` class to represent the basic discrete factor. This class embodies a basic discrete factor (a generalisation of a probability distribution) and its associated factor operators. You will be using this class extensively throughout this course. It provides the basic machinery to automate probabilistic calculations.

Table 1: Discrete Factors (also known as a Discrete Potential Functions / Probability Mass Functions / Conditional PMFs) are represented in `pgmpy` with the `DiscreteFactor` and in `emdw` by the `DiscreteTable` classes. In the following tasks you will use the following member functions – consult the relevant documentation for more details.

| Description                                      | pgmpy                                                     | emdw                                                                                              |
|--------------------------------------------------|-----------------------------------------------------------|---------------------------------------------------------------------------------------------------|
| Multiply two distributions to get their joint    | <code>product</code>                                      | <code>absorb</code>                                                                               |
| Remove a previously multiplied in distribution   | <code>divide</code>                                       | <code>cancel</code>                                                                               |
| Condition on evidence i.e. observe some RVs      | <code>reduce</code>                                       | <code>observeAndReduce</code>                                                                     |
| Marginalize i.e. remove certain RVs from a joint | <code>marginalize</code><br>( <code>spec. remove</code> ) | <code>marginalize</code><br>( <code>spec. keep</code> )                                           |
| Max-marginalize                                  | <code>maximize</code><br>( <code>spec. remove</code> )    | <code>marginalize via</code><br><code>MaxMarginalize</code><br><code>operator (spec. keep)</code> |
| Renormalize a probability table                  | <code>normalize</code>                                    | <code>normalize</code>                                                                            |
| Make a clone of a Factor, using different RVs    |                                                           | <code>copy</code>                                                                                 |

We will specifically avoid using the higher-level magic also available in these packages. We do this to to develop a clear understanding of the basic underlying principles and ideas on which these higher-level constructs are built.

Now code the above coin flipping example using one of these toolboxes. Carefully compare the results with what you have previously obtained using more manual approaches.

## **6 EXPLORE some more: Read Barber chapter 2 – basic graph concepts.**

Next week we expand the use of our factors to be embedded in a network / graphical structure. You might want to take a sneak preview of basic graph concepts.