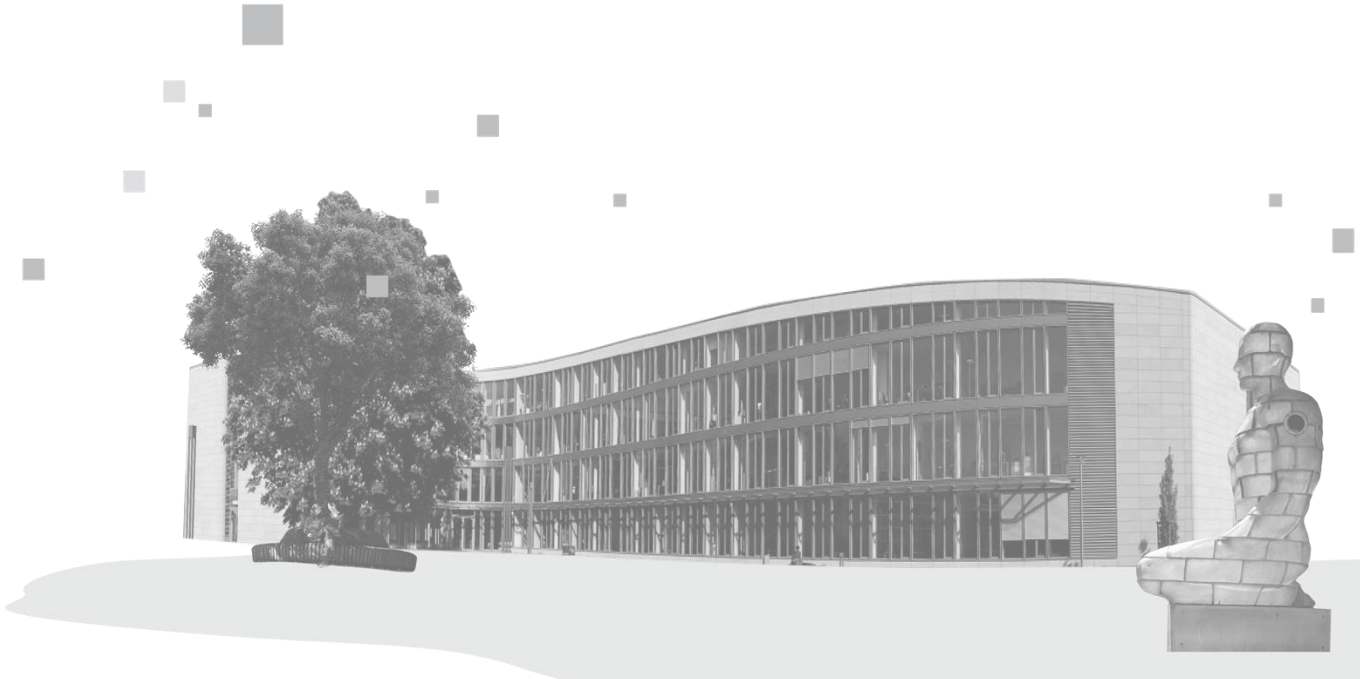
ProbML
Tutorial 22.04.25

Leonhard Hennicke, Rainer Schlosser



3 topics

- LogSumExp trick
- Julia: typed containers and plotting

Logged probabilities

Assignment 1 specifically requires you to store the probabilities of a discrete distribution.

The exercise explicitly asks for logged probabilities to be stored in the distribution. Why?

$$p_1' = \log(p_1)$$

$$p_2' = \log(p_2)$$

How can fundamental operations be implemented?

$$p_1 \cdot p_2 = \exp(p_1' + p_2')$$

$$p_1 + p_2 = \exp(p_1' + \log(1 + \exp(p_2' - p_1')))$$

$$\log(p_1 + p_2) = \log\left(p_1 + \frac{p_2 p_1}{p_1}\right) = \log\left(p_1 + p_1 \exp\left(\log\left(\frac{p_2}{p_1}\right)\right)\right) = \log(p_1 + p_1 \exp(\log(p_2) - \log(p_1)))$$
$$= \log(p_1 \cdot (1 + \exp(\log(p_2) - \log(p_1)))) = \log(p_1) + \log(1 + \exp(\log(p_2) - \log(p_1))) = p_1' + \log(1 + \exp(p_2' - p_1'))$$

Why is this advantageous over the naive approach?

LogSumExpTrick

Assignment 1, Part 1, contains a little function called LogSumExp.

As some of you have already noticed, if you naively implement the logsumexp, some examples given in the comments can not be computed without running into numerical errors.

$$\text{logsumexp}(x_1, \ldots, x_n) = \log(\exp(x_1) + \cdots + \exp(x_n))$$

This little helper is used in various normalization operations and to convert to the stored logprobs back into probabilities.

What is the problem with the formulation found above?

LogSumExpTrick (2)

$$\text{logsumexp}(x_1, \dots, x_n) = y = \log\left(\sum_{k=1}^{n} \exp(x_k)\right)$$

$$\exp(y) = \sum_{k=1}^{n} \exp(x_k) = \exp(c) \cdot \sum_{k=1}^{n} \exp(x_k - c)$$

$$\text{logsumexp}(x_1, \dots, x_n) = c + \log\left(\sum_{k=1}^{n} \exp(x_k - c)\right)$$

c can be chosen arbitrarily, for example using the maximum of the input sequence.

Doing so keeps all exponents at a value smaller than 0, which in turn allows safe computation with arbitrarily large values of x.

Julia typing for containers and plotting

See containers.jl and beta.jl

Hasso Plattner Institut
Digital Engineering · Universität Potsdam

Thank you ☺

Design IT.
Create Knowledge.

www.hpi.de